

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Contribution à l'évolution de la méthode de mesure fonctionnelle COSMIC-FFP techniques d'estimation & processus de certification

Cara, Sébastien

Award date:
2006

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

**Contribution à l'évolution de la méthode
de mesure fonctionnelle COSMIC-FFP:**

**Techniques d'estimation
& processus de certification**

Sébastien Cara

Année académique 2005 – 2006

Mémoire présenté en vue de l'obtention du grade de Maître en Informatique

Table des Matières

I	Introduction et contexte de la méthode de mesure COSMIC-FFP	1
1	Introduction	3
1.1	Historique général des mesures fonctionnelles	3
1.2	La maturité du génie logiciel	5
1.3	Problématiques	6
1.4	Objectifs du mémoire	7
2	Contexte de la méthode de mesure COSMIC-FFP	9
2.1	Introduction	9
2.2	Le modèle du processus de mesure	10
2.3	Validation de COSMIC-FFP	11
2.3.1	1 ^{re} étape du modèle du processus de mesure	11
2.3.2	2 ^e étape du modèle du processus de mesure	13
2.3.3	3 ^e étape du modèle du processus de mesure	15
2.3.4	4 ^e étape du modèle du processus de mesure	15
2.4	Conclusion	15
II	Recherche sur l'estimation de la taille fonctionnelle tôt dans le cycle de développement	17
3	Recherche de techniques d'estimation	19
3.1	Introduction	19
3.2	La technique Early & Quick COSMIC-FFP	20
3.2.1	Critique	22
3.3	Recherche d'une nouvelle technique	22
3.3.1	Méthodologie de l'investigation	22

3.3.2	Outils et techniques	28
3.4	Technique de régression linéaire multiple	29
3.4.1	Le modèle généralisé	29
3.4.2	Transposition du modèle généralisé	30
3.4.3	L'hypothèse H_0	30
3.4.4	Analyse de régression non épurée des variables exogènes	32
3.4.5	Analyse de régression épurée	33
3.4.6	Solution obtenue de la technique de régression linéaire multiple	34
3.4.7	Estimation de l'erreur du modèle	34
3.4.8	Conclusion	36
3.5	Autres techniques	36
3.5.1	Technique des réseaux de neurones	37
3.5.2	Technique de l'arbre de décision <i>repTree</i>	40
3.5.3	Nouvelle définition de l'erreur	42
3.5.4	Technique de l'arbre de décision <i>J48</i> et <i>RandomTree</i>	42
3.5.5	Technique Ripple Down Rule Learner rules	44
3.5.6	Récapitulatif	44
3.6	Conclusion	46

III Opérationnalisation de la certification COSMIC-FFP 47

4 Certification : modèles et approches 49

4.1	La reconnaissance de l'expertise	49
4.2	Modèle de certification de l'IFPUG	50
4.2.1	Description	50
4.2.2	Structure organisationnelle du groupe	51
4.2.3	Procédure de certification	52
4.2.4	Critique	54
4.3	Modèle de la certification du SEI	54
4.3.1	Description	54
4.3.2	Le modèle CMMI	55
4.3.3	Procédure de certification	56
4.3.4	Critique	57

5	Conception du processus de certification pour COSMIC	59
5.1	Introduction	59
5.2	Organisation du processus de certification	60
5.2.1	Deux niveaux de certification	60
5.2.2	Public cible	61
5.2.3	Pré-requis pour l'admission à l'examen	61
5.2.4	Le matériel d'examination	62
5.2.5	Calibrage des questionnaires	63
5.2.6	Validité de la certification	63
5.2.7	Organisation	63
5.3	Logiciel d'administration du processus de la certification	64
5.3.1	Vue d'ensemble	64
5.3.2	Profil des utilisateurs	64
5.3.3	Diagramme des Use Cases	65
5.3.4	Exigences non fonctionnelles	65
5.3.5	Pattern d'architecture	67
5.3.6	Diagramme de classe	68
5.3.7	Diagrammes de séquence	70
5.4	Comparaison avec l'existant	73
5.5	Conclusion	74
IV	4^ePartie : Conclusion et perspectives à venir	75
6	Conclusion	77
7	Perspectives à venir	79
A	Scénarii relatifs aux cas d'utilisation	85
A.1	UC1 Login	85
A.2	UC2 Ajout d'un candidat	85
A.3	UC3 Ajout d'un administrateur/examineur	86
A.4	UC4 Modification d'un mot de passe	87
A.5	UC5 Sélection d'un candidat	88
A.6	UC6 Sélection d'un examen	89
A.7	UC7 Inscription à un examen	89
A.8	UC8 Ajout d'un résultat	90

A.9	UC9 Consultation des inscriptions	90
A.10	UC10 Impression des inscriptions dans un fichier pdf	91
A.11	UC11 Suppression d'un examen	91
A.12	UC12 Ajout d'un examen	92
A.13	UC13 Sélection d'une personne	92
A.14	UC14 Suppression d'une personne	93
A.15	UC15 Modification d'une personne	93
A.16	Dictionnaire des Use Cases	94
B	Comparatif des résultats	95
C	Captures d'écran du logiciel	99
C.1	Identification de l'administrateur	99
C.2	Interface principale	100
C.3	Interface d'inscription	101
C.4	Interface d'attribution d'une note	102
C.5	Interface de suppression d'un examen	103

Liste des Tableaux

3.1	Intervalles numériques pour les catégories et niveaux	21
3.2	Echantillon d'analyse	27
3.3	Résultat de la régression.	32
3.4	Tableau de rejet de l'hypothèse nulle	33
3.5	Résultat de la seconde régression.	33
3.6	Variance	35
B.1	Valeurs estimées par les différentes techniques d'estimation	97
B.2	Récapitulatif des estimations par application	98

Liste des Figures

1.1	Historique des mesures	4
1.2	Statistiques d'achèvement de projets informatiques	6
2.1	Measurement Process – High-Level Model [13]	10
2.2	Measurement Process – first step detailed	11
2.3	Métamodèle de COSMIC-FFP [12]	12
2.4	Measurement Process – second step detailed	13
2.5	FUR extraction [21]	14
3.1	Processus de décomposition hiérarchique E&QCFFP	21
3.2	Différents niveaux d'abstraction de la phase de développement	24
3.3	Modélisation du processus à atteindre	28
3.4	Zone d'acceptation d' H_0	31
3.5	Visualisation du plan de régression	34
3.6	Structure d'un neurone humain	38
3.7	Structure d'un neurone artificiel[47]	38
3.8	Les trois types de fonction d'activation	38
3.9	Modèle du réseau de neurones	39
3.10	Représentation du modèle de réseau de neurones	39
3.11	Arbre de décision	40
3.12	Arbre de décision élagué	41
3.13	Arbre de décision J48	42
3.14	Algorithme du RandomTree	43
3.15	Ripple Down Rule Learner rules	44
3.16	Erreur : régression	45
3.17	Erreur : réseau de neurones	45
3.18	Erreur : arbre repTree	45
3.19	Erreur : arbre RandomTree	45

3.20	Erreur : arbre j48	45
3.21	Erreur : règles RIDOR	45
4.1	Les différents niveaux de certification CMMI	56
4.2	Phases d'évaluation SCAMPI	56
5.1	Principales raisons de se certifier[39]	60
5.2	Principaux avantages retirés d'une certification[39]	60
5.3	Structure de l'examination	63
5.4	Use Cases Diagram	66
5.5	MVC pattern[33]	68
5.6	Diagramme de classe	69
5.7	Diagramme de séquence d'ajout d'un candidat	71
5.8	Diagramme de séquence d'ajout d'un nouvel examen	72
5.9	Diagramme de séquence d'ajout d'une inscription	73
C.1	Interface d'identification	99
C.2	Interface principale	100
C.3	Interface d'inscription à un examen	101
C.4	Interface d'attribution d'une note	102
C.5	Interface de suppression d'un examen	103

Résumé

Les travaux et recherches autour de la méthode de mesure fonctionnelle *COSMIC-FFP* ne cessent de se poursuivre depuis sa publication en tant que norme ISO 19761 : 2003. Son innovation au sein des mesures fonctionnelles explique l'engouement lié à son développement et à sa reconnaissance internationale.

C'est dans ce contexte que ce mémoire rend compte, dans un premier temps, des récentes recherches de modèles permettant d'obtenir une estimation de la taille fonctionnelle d'une application sans avoir à réaliser complètement tout le processus de mesure. L'intérêt de cette recherche est évident étant donné la situation économique actuelle.

Dans un deuxième temps, il sera question du processus de certification de la mesure *COSMIC-FFP*. Celle-ci procurera aux mesureurs la reconnaissance de leur compétence. Après une analyse des certifications existantes, la transparence sur le développement de l'organisation de la certification de *COSMIC-FFP* permettra une évaluation à la fois de la qualité et de l'état d'avancement.

Abstract

Works and researches on the functional sizing method *COSMIC-FFP* have constantly kept going since its publication as the international standard ISO/IEC 19761:2003. Its innovative nature within functional sizings accounts for its international recognition and the infatuation for its development.

In this frame, this thesis will, first of all, give an account of recent researches of patterns allowing to obtain a reckoning of a mapping functional size without having to make the whole sizing process. The relevance of this research is obvious, given the current economic situation. This thesis will then deal with the sizing process *COSMIC-FFP*'s certification which would give the metric specialists a recognition of their competences. After an analysis of the existing certifications, the transparency on the development of the *COSMIC-FFP* certification's organization will allow an estimation of its quality and state of advancement.

Avant-Propos

Ce document est l'aboutissant de mon stage réalisé à l'ETS (Ecole de Technologie Supérieure) de Montréal. Ce stage m'a permis de découvrir un champs du génie logiciel que j'ai pu approfondir afin de réaliser ce mémoire.

La rédaction de ce mémoire n'est pas le fruit de mon seul labeur. C'est pourquoi je remercie toutes les personnes qui, de près ou de loin, ont contribué la réalisation de ce travail.

Je tiens tout d'abord à remercier mon promoteur, le Professeur Naji Habra, pour son soutien et ses conseils judicieux qu'il m'a procurés tout au long de ce mémoire.

Je remercie également mon maître de stage, le Professeur Alain Abran, précurseur de la méthode de mesure fonctionnelle COSMIC-FFP. En plus de son chaleureux accueil au Canada, il m'a été d'une aide très précieuse.

Finalement, je tiens à remercier tous mes proches et plus particulièrement mes parents, auxquels je dédie ce mémoire, pour m'avoir permis de réaliser mes études.

Première partie

Introduction et contexte de la méthode de mesure COSMIC-FFP

Chapitre 1

Introduction

1.1 Historique général des mesures fonctionnelles

La mesure de la taille fonctionnelle en génie logiciel n'est en réalité encore qu'à ses débuts. Cette jeune discipline a pris ses racines en 1979 avec Allan Albrech travaillant pour IBM. Jusque là, les quelques mesures réalisées n'étaient que purement physiques ou directes [45]. Elles portaient sur le coût, l'effort, les lignes de code, la vitesse, la taille mémoire ou encore le nombre d'erreurs. Son projet au sein d'IBM fut de trouver une mesure capable d'aider à évaluer la productivité d'une organisation.

The purpose...is to provide each AD/M [Application Development and Maintenance, PV] site a consistent way to measure, portray and demonstrate the productivity of their AD/M activities. [20]

Il eut l'idée d'évaluer la taille d'un logiciel en ne se référant qu'à ses fonctionnalités d'un point de vue utilisateur. Cette approche est nommée FPA pour Functional Point Analysis [19]. Les principaux buts visés par sa mesure sont :

- d'être indépendant de la technologie utilisée pour le développement de l'application.
- d'être simple d'utilisation.
- de pouvoir être relativement bien estimé à partir de la spécification des exigences.
- d'être significatif pour l'utilisateur.

Comme illustré à la figure 1.1, cette méthode a évolué en suivant différentes directions parallèles :

- En 1986 a été créée une organisation nommée *IFPUG* (International Function Point Users' Group) afin d'assurer notamment une plus grande diffusion. Un manuel [28] a été publié afin de fournir une base standardisée des règles et du fonctionnement de la méthode FPA.
- En 1986, la méthode *Feature Point* [29] avait pour but d'appliquer la méthode FPA

aux algorithmes scientifiques. La taille de l'application croît proportionnellement à la complexité de l'algorithme. Cependant, étant donné la complexité intrinsèque de calcul de tels algorithmes, elle fut relativement rapidement abandonnée.

- En 1988, la méthode *MkII Function Point Method* [41] améliore l'approche d'Albrecht en considérant les logiciels de gestion très riches en données.
- En 1992, la méthode *3D Function Points* [46] adaptant la méthode FPA au logiciel en temps réel.

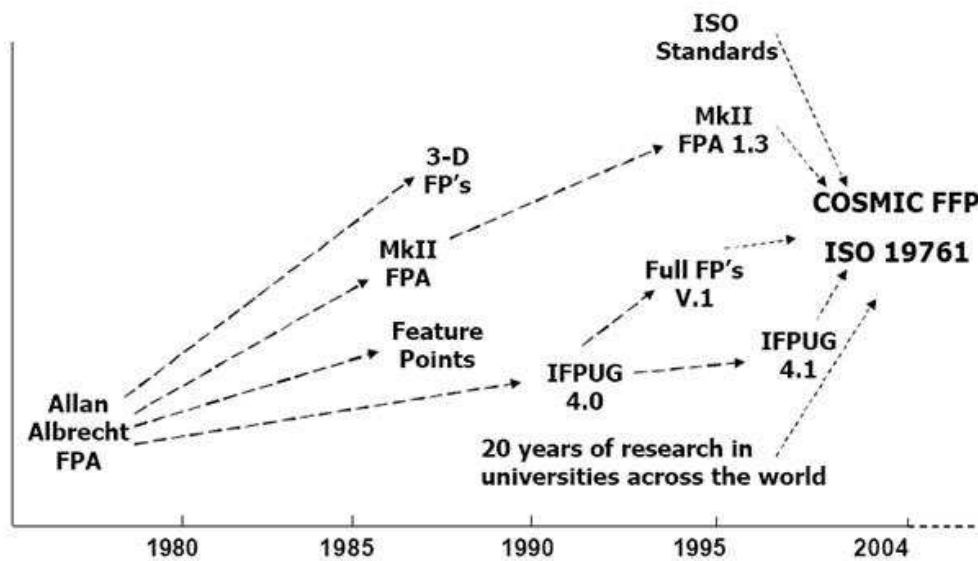


FIG. 1.1 – Historique des mesures
[16]

Une nouvelle méthode est également apparue tirée directement des règles IFPUG ; il s'agit de *Full Function Point Method* [37]. Elle ajoute plusieurs composants afin de calculer la taille de logiciel en temps réel.

L'un des problèmes à l'époque résidait dans l'évaluation de la qualité de la mesure fonctionnelle ; laquelle choisir ? Un groupe de travail fut créé sous ISO/IEC JTC1/SC7 en 1992 afin d'établir une série de standards donnant les principes directeurs à respecter pour une mesure fonctionnelle. Un rapport technique fut alors rédigé et publié sous la forme de la norme ISO 14143 [2] permettant de vérifier et d'évaluer des méthodes de mesure fonctionnelle.

Sur la base de cette expérience, une nouvelle méthode a été conçue en ayant pour objectif d'être compatible avec la norme ISO 14143. Elle fut créée de façon à pouvoir être appliquée tout aussi bien sur des applications de gestion d'entreprise telles que des banques, assurances ou

encore les systèmes de gestion de personnel que sur des logiciels en temps réel. Les concepteurs se sont regroupés en 1998 sous l'organisation COSMIC pour *Common Software Measurement International Consortium*. Une première version de la méthode *COSMIC-FFP* fut publiée en 1999. Depuis, elle a été reconnue officiellement sous la norme ISO 19761 :2003 [3].

1.2 La maturité du génie logiciel

L'informatique joue un rôle de plus en plus primordial dans notre société. Elle est omniprésente et son invasion dans la vie quotidienne est constante. Il est donc essentiel de fournir à tous ces systèmes un logiciel de qualité afin d'éviter que se réitèrent de nombreux problèmes ayant marqué les esprits. Les trois exemples ci-dessous sont un échantillon de pannes informatiques déjà rencontrées :

- **Le réseau de la SNCF paralysé** [38]

Une panne informatique a paralysé quelque 800 terminaux de vente sur 4.400. Cette défaillance est due à un algorithme défectueux qui a progressivement contaminé les terminaux de vente en gare.

- **un Missile *ami*** [17]

Un navire de guerre anglais fut coulé par un Exocet français au cours de la guerre des Malouines. Le vaisseau anglais n'avait en fait pas activé ses défenses et n'a rien tenté pour se protéger. Plusieurs centaines de morts furent déplorés. La cause n'est autre que l'Exocet qui n'était pas répertorié comme missile ennemi.

- **Echec du premier lancement d'Ariane V** [26] Au premier lancement de la fusée Ariane V, celle-ci a explosé en vol. La cause provenait du logiciel de plate-forme inertielle repris tel quel d'Ariane IV sans nouvelle validation.

Comme le dénombre le plus récent rapport de l'étude CHAOS [42] en 2004, bon nombre de projets informatiques connaissent encore des problèmes.

Seuls 29% des projets informatiques sont achevés à temps et dans le budget fixé alors que 18% sont totalement abandonnés. Les projets faisant partie des 53% restant se sont soit terminés trop tard, hors budget et/ou seulement partiellement accomplis.

Il est certain que des mesures fiables de la taille du logiciel, réalisée à partir du cahier de charge contribueraient à faire avancer la maturité du génie logiciel. L'évaluation de la taille d'une application à réaliser fournit au responsable du projet un outil fiable pour la négociation du budget et du délai nécessaire à la réalisation de l'application. Par ailleurs, ces mesures peuvent

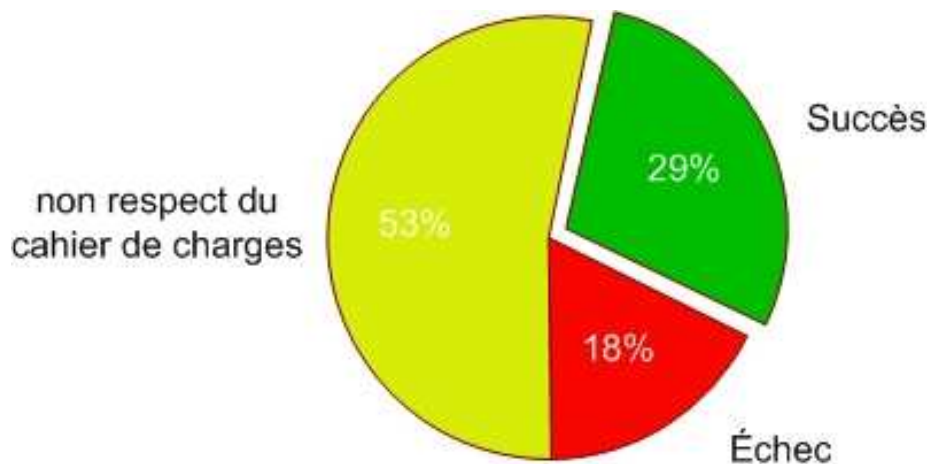


FIG. 1.2 – Statistiques d’achèvement de projets informatiques
[16]

être utilisées afin d’évaluer la productivité de l’équipe responsable du projet. C’est dans cette optique que la méthode de mesure fonctionnelle COSMIC-FFP a été conçue. Toutefois, ces mesures ne solutionneront pas le problème dans son ensemble.

1.3 Problématiques

Comme énoncé dans la section précédente, l’un des problèmes structurels liés aux projets informatiques est inhérent à la gestion de projet et plus particulièrement à l’estimation des délais et des coûts. La nécessité d’employer des techniques de mesure évaluant l’application afin de pallier ces gros problèmes se fait de plus en plus ressentir.

Cependant, la société capitaliste pousse à réduire un maximum les coûts jugés non indispensables tels que ceux liés à une estimation de qualité de la taille d’une application. Ces négligences peuvent handicaper le projet lorsque par exemple sa durée est sous-estimée. Ce cas de figure permet d’envisager plusieurs scénarii tels que l’abandon pur et simple du projet, sa livraison hors délais ou encore à temps mais nécessitant dans ce cas le développement de mises à jour (patch). Les coûts sont alors bien plus importants que ceux qui auraient permis une bonne planification.

Cependant, la mesure d’une application n’est pas une condition suffisante à la réussite d’un projet. Il se peut en effet que la mesure n’ait pas été effectuée dans les règles de l’art. La mauvaise qualité d’une mesure peut tout aussi bien être catastrophique pour le projet, menant

à une mauvaise estimation des délais ou du budget.

1.4 Objectifs du mémoire

Ce mémoire a pour objectif de contribuer à faire avancer la mesure COSMIC-FFP afin d'apporter une solution à la problématique décrite ci-dessus. Celle-ci sera traitée en deux parties distinctes.

Dans un premier temps, il sera question d'arriver à un bon compromis sur le coût lié au processus de mesurage fonctionnel des applications confortant la budgétisation et la planification du projet. A cette fin, il sera question de recherches de ratios permettant d'estimer avec précision la taille fonctionnelle d'une application sans réaliser l'ensemble du processus de mesure. En d'autres termes, une estimation de la taille de l'application pourra être obtenue sans disposer de toutes les informations relatives à l'application mesurée. Un éventuel résultat permettrait donc une économie et un gain de temps tout en permettant une gestion plus sûre de projet.

La seconde partie de ce mémoire traitera du processus de certification de la norme ISO 19761. En effet, nonobstant la grande qualité de la norme, l'application de celle-ci par des personnes néophytes pourrait avoir un impact sur la précision des mesures. C'est donc dans le but de pallier ces imprécisions de mesures qu'a été lancé le projet de certification des mesureurs utilisant la méthode de mesure COSMIC-FFP. Les entreprises pourront également avoir une plus grande certitude quant à la qualité des mesures COSMIC de leurs projets ; ce qui devrait consolider l'utilisation de COSMIC-FFP. Ainsi, un projet correctement évalué sur base des FURs (*Functionality User Requirement*) donnera une bien meilleure mesure de la taille de l'application ainsi que du temps imparti à une équipe pour arriver à ses fins.

Afin de pouvoir développer ces sujets, ce mémoire est agencé selon la structure suivante. Un premier chapitre introduira brièvement les principes essentiels de la méthode de mesure fonctionnelle COSMIC-FFP. Il sera question ensuite de la recherche sur l'estimation de la taille fonctionnelle d'une application tôt dans le cycle de développement. La partie suivante débutera par un chapitre décrivant les modèles actuels de certification afin de cerner l'existant. On disposera alors des outils permettant de développer le chapitre consacré au processus de la certification de COSMIC-FFP. Finalement, la conclusion rendra compte des objectifs atteints à travers ce mémoire et lancera quelques pistes sur les perspectives.

Chapitre 2

Contexte de la méthode de mesure COSMIC-FFP

2.1 Introduction

« When you can measure what you are speaking about, and express it in numbers, you know something about it. But when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind [30]. »

Les méthodes de mesure sont donc essentielles pour la reconnaissance du génie logiciel comme ingénierie. Dès lors, beaucoup de chercheurs ont oeuvré à cette fin de reconnaissance depuis 1970 sur les méthodes de mesure logicielle.

L'objectif premier d'une méthode de mesure dans le génie logiciel est d'essayer de donner une idée de la taille la plus représentative possible d'un logiciel. Enormément de mesures ont été envisagées comme par exemple celle qui consiste à compter le nombre de lignes de code. Cependant, ces méthodes sont loin d'être représentatives : compter le nombre de lignes de code d'un logiciel peut donner un taux d'erreurs allant jusqu'à cinq cents pour cent !

Les méthodes de mesure fonctionnelles ont su s'imposer à travers cette anarchie. Ces dernières sont très utiles puisqu'elles sont dérivées directement des spécifications très tôt dans le cycle de développement du logiciel. Elles permettent ainsi d'obtenir une idée globale de la taille de l'application et facilitent la planification du projet.

Ce chapitre a pour ambition d'introduire COSMIC-FFP qui fait partie de la seconde génération

des méthodes de mesure fonctionnelle. Cette méthode de mesure se veut être, depuis sa création, d'une grande qualité. A cette fin, elle s'est conformée à la norme ISO 14143 [2] spécifiant les critères de design nécessaires à la reconnaissance en tant que norme internationale pour les méthodes de mesure de la taille fonctionnelle.

Ainsi, il sera question tout d'abord du modèle du processus de mesure [13] d'Alain Abran et Jean-Philippe Jacquet qui servira de fil conducteur pour la présentation de COSMIC-FFP. Ensuite, une attention particulière sera portée sur l'état des recherches actuelles sur COSMIC-FFP par le biais de ratios permettant d'obtenir très tôt une estimation de la taille logicielle dans le cycle de développement.

2.2 Le modèle du processus de mesure

Nombre de modèles du processus de mesure ont été inventés dans le but de valider une méthode de mesure. Nonobstant cette profusion, qu'il soit mathématique ou plus pratique, aucun d'eux n'a pu rencontrer l'unanimité quant à son acceptation. C'est dans cet environnement que messieurs Abran et Jacquet ont créé leur propre modèle de mesure.

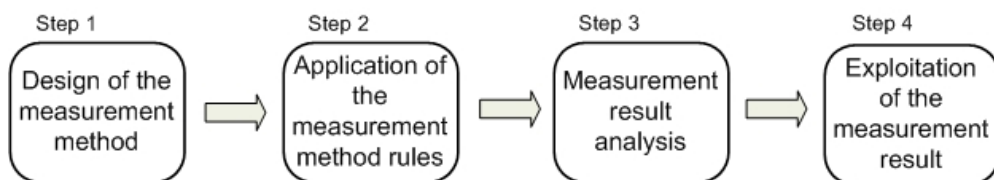


FIG. 2.1 – Measurement Process – High-Level Model [13]

Comme le montre le schéma ci-dessus, ce modèle est divisé en quatre étapes :

- 1^{ère} étape : avant de mesurer, il est nécessaire de concevoir une méthode de mesure.
- 2^{ème} étape : les règles d'une méthode de mesure sont appliquées sur le logiciel ou la partie de celui-ci devant être mesurée.
- 3^{ème} étape : l'application des règles de la méthode de mesure produit des résultats.
- 4^{ème} étape : les résultats obtenus peuvent être analysés et exploités dans un modèle qualitatif ou quantitatif.

2.3 Validation de COSMIC-FFP

2.3.1 1^{re} étape du modèle du processus de mesure

La première étape consiste à concevoir la méthode de mesure. Elle est subdivisée en quatre sous-étapes. Le schéma suivant les reprend avec leurs interactions.

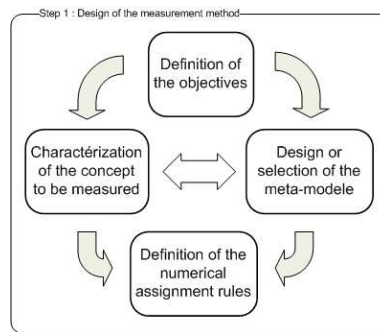


FIG. 2.2 – Measurement Process – first step detailed

La définition des objectifs

La méthode de mesure COSMIC-FFP avait pour objectif principal de vouloir s'appliquer à trois types de logiciel [3] :

- Les logiciels de gestion à caractère généralement administratif
- Les logiciels en temps-réel interagissant avec le monde réel tels que les ascenseurs
- Les logiciels hybrides des deux types précédents

Par ailleurs, COSMIC-FFP se veut de respecter intégralement la norme ISO 14143 [1] définissant les concepts d'une méthode de mesure fonctionnelle. Cela lui permet ainsi d'être compatible avec les concepts modernes de l'ingénierie logicielle. Cette méthode de mesure a également l'avantage de laisser le choix quant au type de point de vue de mesure adopté (utilisateur ou développeur [21]).

Métamodèle de COSMIC-FFP

Le métamodèle de COMSIC-FFP donne la modélisation du logiciel sur lequel la méthode de mesure est appliquée. Selon cette vue, l'utilisateur est séparé du système par une frontière. Les deux interactions possibles à travers cette frontière sont les « *entrées* » et les « *sorties* ». Cela fait l'objet de la première règle principale [3] :

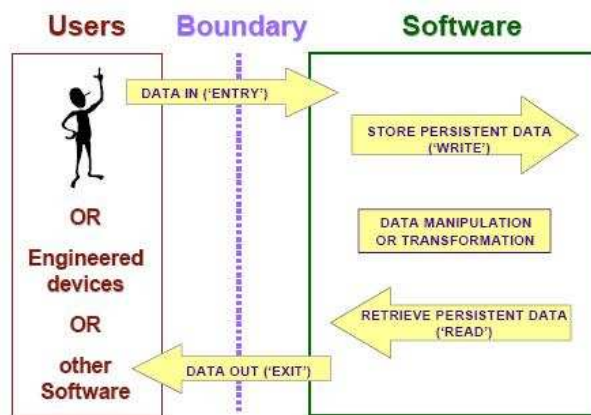


FIG. 2.3 – Métamodèle de COSMIC-FFP [12]

« *The software to be mapped and measured is fed by input and produces useful output, or a useful outcome, to users.* »

Lorsque le système a besoin de lire ou de sauvegarder une donnée stockée sur une base de données ou dans un fichier local, il y accède respectivement au moyen d'une « *lecture* » ou d'une « *écriture* ». Toutes les règles permettant d'identifier ces concepts sont explicitées dans le guide de mesure COSMIC [21]. Comme il sera mentionné plus tard, ce sont ces quatre types de mouvements de données qui sont comptabilisés.

Caractérisation du concept mesuré

Le concept mesuré fait l'objet de la seconde règle principale [3] :

« *The software to be mapped and measured manipulates pieces of information designated as data groups which consist of data attributes.* »

Ces mouvements de données sont définis dans la norme comme suit :

« *A Base Functional Component which moves one or more data attributes belonging to a single data group* » [3]

Cette notion de groupe de données¹ a été introduite afin d'éviter tout problème lié à l'abstraction du mesureur. C'est pour cette raison que l'unité étalon de la mesure COSMIC n'est pas le processus fonctionnel, celui-ci étant trop dépendant du mesureur. C'est le même prin-

¹Ensemble distinct non vide, non ordonné et non redondant de types d'attributs. Un attribut se définit par la plus petite parcelle d'information codée, dans un groupe de données, possédant une signification dans la perspective des Fonctionnalités Utilisateurs Requises (FUR).

cipe pour les use cases, ceux-ci sont dépendants du niveau d'abstraction que veut lui donner l'analyste.

Définition des règles d'assignation numérique

Une fois tous les groupes de données identifiés, la mesure en tant que telle peut commencer. Pour chaque processus fonctionnel (PF) identifié, le mesureur identifie chaque mouvement de données. Ces derniers lui permettent d'appliquer la définition suivante lui donnant pour résultat la taille fonctionnelle de l'application [3] :

$$\begin{aligned} Taille_{Cfsu}(processus\ fonctionnel_i) = & \sum taille(entrees_i) + \sum taille(sorties_i) \\ & + \sum taille(lectures_i) + \sum taille(ecritures_i) \end{aligned}$$

Par convention, les quatre types de mouvement de données ont la même valeur : une unité COSMIC (1 Cfsu). Comme il s'agit d'une méthode de mesure fonctionnelle, il convient de mesurer le type de mouvement de données et non ses occurrences. Par exemple, si une lecture d'un fichier requiert plusieurs accès à celui-ci, en pratique un seul sera compté.

Ceci n'est qu'un bref aperçu du fonctionnement de la méthode COSMIC-FFP. De nombreuses autres règles sont reprises dans la norme ISO/IEC 19761 :2003 [3] et dans le guide COSMIC [21].

2.3.2 2^e étape du modèle du processus de mesure

La deuxième étape du modèle du processus de mesure consiste à appliquer la mesure préalablement définie sur un cas concret. La figure 2.4 illustre la démarche.

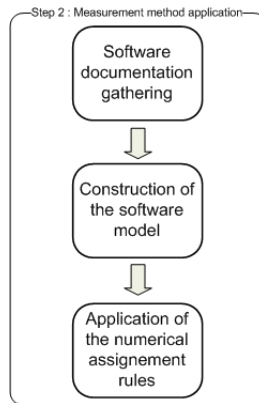


FIG. 2.4 – Measurement Process – second step detailed

Documentation logicielle

Cette sous-section consiste à récolter l'information nécessaire d'un logiciel afin d'en mesurer ses fonctionnalités. Dans la perspective de mesure COSMIC-FFP, cette information est appelée « Fonctionnalités Utilisateurs Requise » (FUR). Elles peuvent être obtenues de différentes manières, antérieures ou postérieures à l'implémentation comme illustré dans la figure 2.5. L'extraction des FUR fournit, indépendamment de la source utilisée, un modèle ne contenant que les fonctionnalités proposées aux utilisateurs.

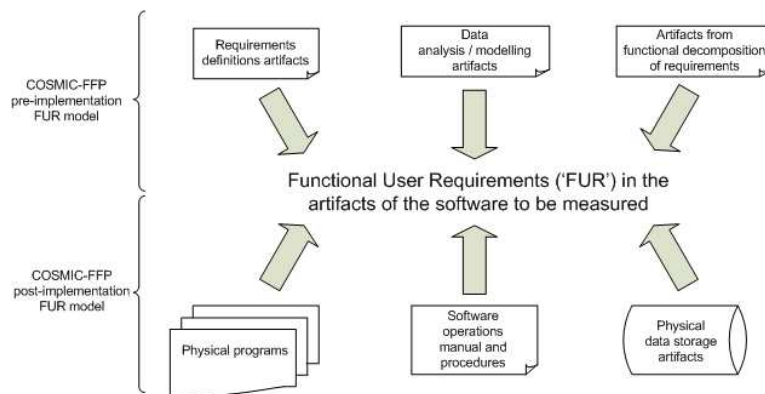


FIG. 2.5 – FUR extraction [21]

Construction du modèle logiciel

La construction du modèle logiciel a pour but de décrire comment le logiciel mesuré est représenté selon le métamodèle de la méthode de mesure. Dans la méthode COSMIC-FFP, le modèle logiciel est le résultat de la phase dite « de mise en correspondance COSMIC-FFP ». Celle-ci applique le métamodèle de la figure 2.3 au logiciel donné.

Dans un premier temps, les différents niveaux d'abstraction des FURs sont identifiés, il s'agit des couches logicielles. Ensuite, au sein de chaque couche, la frontière est identifiée, distinguant ainsi les items faisant partie du logiciel (à l'intérieur de la frontière) et ceux appartenant à l'environnement opérationnel du logiciel (hors de la frontière). C'est également dans cette étape que les utilisateurs interagissant avec les processus fonctionnels, les groupes de données et les événements déclencheurs sont identifiés. Finalement, la partie la plus importante de la mise en correspondance détermine les mouvements de données pour chaque processus fonctionnel.

Assignation des règles numériques

Une fois le modèle logiciel établi, la mesure proprement dite peut être effectuée à travers « la phase de mesure COSMIC-FFP » sur base du principe suivant [21] :

« La taille fonctionnelle du logiciel est directement proportionnelle au nombre de ses mouvements de données ».

La norme de mesure de 1 Cfsu (Cosmic Functional Size Unit) est définie par convention comme étant équivalente à un seul mouvement de données. La taille fonctionnelle du logiciel peut ainsi être calculée en suivant tous les principes érigés dans le manuel de mesure COSMIC.

2.3.3 3^e étape du modèle du processus de mesure

Des résultats bruts n'ont en soi guère de sens, c'est pour cette raison que ces résultats vont être étoffés de diverses informations telles que la méthode utilisée, sa version, la description du processus de mesures, ... Ces informations vont ainsi permettre de donner un sens à la valeur de la mesure obtenue.

La documentation et l'archivage des résultats provenant de la méthode de mesure COSMIC-FFP sont régis par les règles du cinquième chapitre du manuel de mesures COSMIC [21].

2.3.4 4^e étape du modèle du processus de mesure

La dernière étape du processus de mesure regroupe toutes les activités exploitant les résultats récoltés lors de la troisième étape. On y retrouve divers modèles tels que ceux de qualité, de productivité ou de budget.

L'une des analyses les plus intéressantes et utiles du point de vue gestion de projet détermine, avec un taux d'erreur relativement faible, la durée effective que devrait prendre un projet pour sa réalisation pour une équipe donnée. La recherche de ratios faisant l'objet du chapitre suivant s'insère dans le cadre de cette étape du processus de mesure.

2.4 Conclusion

Ce paragraphe destiné à donner une présentation originale sur la méthode de mesure fonctionnelle COSMIC-FFP s'est voulu de rester général. La méthode est bien plus précise sur de nombreux aspects.

Cette brève introduction permet la transition vers les recherches et travaux actuels qui font l'objet du chapitre suivant. Il sera question dans un premier temps de trouver une manière d'obtenir une bonne estimation de la taille fonctionnelle d'une application très tôt dans sa phase de développement.

Deuxième partie

Recherche sur l'estimation de la taille fonctionnelle tôt dans le cycle de développement

Chapitre 3

Recherche de techniques d'estimation

3.1 Introduction

La norme COSMIC-FFP n'a pas encore livré tous ses secrets. Pas mal de recherches restent en effet à réaliser. Nonobstant une certaine maturité et sa reconnaissance en tant que norme — ISO 19761 :2003 [3] —, elle fait ponctuellement l'objet de mises à jour officielles appelées « *Update Bulletins* » afin de préciser certaines notions. Le statut conféré à ces mises à jour est égal à celui de la norme dans le sens où elles peuvent être interprétées comme une extension officielle d'ISO 19761 :2003. A l'heure actuelle, deux mises à jour sont parues : la première traitant des objets d'intérêts et la seconde précisant le concept de couches logicielles.

Par ailleurs, un grand nombre de travaux de recherche, outre celui de certification, portent directement ou non sur la méthode de mesure fonctionnelle COSMIC-FFP. C'est le cas d'une étude menée conjointement avec le professeur A. Abran (Ecole de Technologie Supérieure, Canada) ayant pour objectif la recherche de ratios permettant *d'estimer* la taille fonctionnelle d'une application dans les phases préliminaires du cycle de développement. Avoir connaissance d'une estimation valable de la taille d'une application lors de la réalisation de son cahier des charges contribue réellement à la justesse de celui-ci. Les risques de rendre une application hors délai ou hors budget, comme il en est encore trop souvent question [42], sont dès lors revus à la baisse à la plus grande satisfaction du client et de l'entreprise.

En outre, les spécifications fonctionnelles sont souvent finalisées assez tard dans le processus de développement alors qu'une part significative du budget risque déjà d'être dépensée. Il est toutefois possible d'obtenir la taille fonctionnelle sans posséder toutes les spécifications fonctionnelles ; dans ce cas il faut accepter une plus grande imprécision de la taille logicielle. Le

coût lié à cette phase de mesure se trouve ainsi réduit ; le problème étant alors de trouver le seuil minimal d'information nécessaire à une mesure représentative et exploitable.

Cette section du mémoire va dans un premier temps mettre en avant certaines méthodes d'estimation qui ont d'ores et déjà pu être développées comme c'est le cas de la technique *Early & Quick COSMIC-FFP* [40]. Ensuite, une attention particulière sera accordée aux récents travaux de recherche menés avec l'ETS sur les ratios susmentionnés. A cette fin, la méthode de recherche utilisée pour les découvrir et les résultats obtenus seront développés.

3.2 La technique Early & Quick COSMIC-FFP

La technique d'estimation du *Early & Quick COSMIC-FFP* a le gros avantage d'être, comme son nom l'indique, très simple et de fournir rapidement une estimation de la taille fonctionnelle d'un logiciel. L'idée est de catégoriser chaque processus fonctionnel afin de lui attribuer une valeur pré-déterminée.

Tout d'abord, cette méthode requiert la construction d'une structure hiérarchique des fonctionnalités sous forme d'arbre à partir des spécifications fonctionnelles du logiciel. Cette arbre doit avoir la structure illustrée par la figure 3.1. La racine (ou point de départ) est généralement le niveau du logiciel en entier puis on descend dans l'arbre en attribuant à chaque noeud un nom en relation avec les fonctions qu'il regroupe. Le dernier niveau est atteint lorsque le mesureur juge qu'il n'est plus nécessaire de décomposer davantage. Cet arbre peut dès lors être plus ou moins précis selon sa hauteur.

Une propriété importante repose dans le fait que toutes les fonctionnalités de base de l'application doivent se trouver dans les feuilles, cela sous-entend qu'une fonctionnalité d'un noeud doit impérativement être représentée explicitement dans une feuille héritée de ce noeud. Cela implique donc que si on a la mesure de chaque feuille, on a la mesure de l'arbre tout entier et donc de l'application. Dès lors, la qualité de l'estimation de la mesure dépend en grande partie de l'aptitude du mesureur à bien catégoriser les fonctions au sein de son arbre (i.e. General Process ou Macro Process). En d'autres termes, il doit être capable d'estimer à quel niveau de la topologie appartient un élément du logiciel.

Une fois le niveau déterminé, le mesureur endosse la responsabilité d'attribuer arbitrairement à chaque fonction la catégorie à laquelle il appartient selon le tableau 3.1. On peut noter que l'intervalle de la catégorie sélectionnée peut être très large, à plus forte raison lorsque l'on se

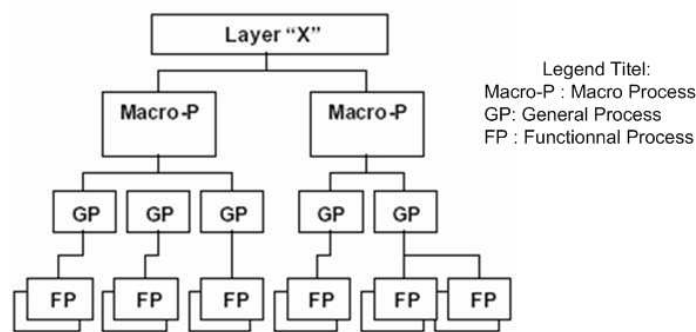


FIG. 3.1 – Processus de décomposition hiérarchique E&QCFFP

situé au niveau des Macro Process, ce qui peut ainsi conduire à de grosses variations. Une fois la catégorie sélectionnée, il suffit d'associer chaque processus fonctionnel à leur valeur respective¹ puis de les sommer afin d'obtenir une estimation de la taille fonctionnelle du logiciel.

Ce tableau a été conçu sur base d'analyse statistique d'un ensemble de données. Il offre pour chaque niveau une valeur minimale, moyenne et maximale.

Niveau	Catégorie	estimation
Processus Fonctionnel	Petit	n1 Cfsu
	Moyen	n2 Cfsu
	Grand	n3 Cfsu
Processus Général	Petit	6 – 12 FP
	Moyen	13 – 19 FP
	Grand	20 – 25 FP
Macro Processus	Petit	2 – 3 GP
	Moyen	4 – 7 GP
	Grand	8 – 12 GP

TAB. 3.1 – Intervalles numériques pour les catégories et niveaux

Une fonction est toujours classée dans l'un des trois niveaux d'abstraction suivant :

- Un processus fonctionnel (FP) : même définition que dans la norme ISO 19761 [3] ; c'est-à-dire le plus petit processus permettant à l'utilisateur d'atteindre un objectif à un niveau opérationnel.
- Un processus général (GP) : un ensemble de processus fonctionnels liés à un sous-système opérationnel de l'application visant à répondre à un besoin de l'utilisateur.

¹les valeurs n1, n2 et n3 doivent être trouvées de manière empirique sur le nombre de cas étudiés

- Un macro processus (MP) : peut constituer de temps en temps un projet de développement en entier. C'est un sous-ensemble pertinent de la totalité d'un système d'information.

Les trois valeurs du tableau 3.1 correspondant aux processus fonctionnels sont laissés à l'appréciation du mesureur.

3.2.1 Critique

Ce genre de technique a cependant du mal à se justifier formellement. Il relève d'une part d'une subjectivité assez importante, basée sur l'expérience du mesureur, pour pouvoir la mettre en oeuvre. D'autre part, lors de toute estimation, le taux d'erreur permet d'évaluer la qualité de l'estimation par rapport à la réalité. Or, la technique *Early & Quick COSMIC-FFP* n'offre pas de moyen d'estimation de l'erreur. Le mesureur ne peut dès lors jamais donner d'indication sur la qualité de son estimation.

3.3 Recherche d'une nouvelle technique

3.3.1 Méthodologie de l'investigation

La méthodologie de recherche employée se veut être la plus rigoureuse possible. La part de subjectivité relative à l'usage de certains procédés tels que **Early & Quick COSMIC-FFP** ne peut être admise. C'est pourquoi il sera fait usage d'outils scientifiques permettant d'apporter plus de précision dans la mesure en laissant au maximum de côté la subjectivité du mesureur.

Cas d'étude

Le matériel employé pour effectuer ces travaux sont huit cas d'application [9, 6, 10, 7, 8, 11, 5, 4] représentatifs qui ont été étudiés par Adel Khelifi, Alain Abran, Charles Simons et Jean-Marc Desharnais. Ces cas ont été construits de façon rigoureuse conformément à la norme ISO 19761 [3]. En outre, ils font tous l'objet d'une documentation détaillée dans le rapport technique ISO/IEC TR 14143-4 (Version 2000). Ce document ISO fournit un choix varié de *Functional Users Requirements* (FUR) dont l'objectif est de mettre à disposition des chercheurs et praticiens un matériel de qualité pour leurs travaux.

Ces cas forment un échantillon idéal représenté dans le tableau 3.2 pour une analyse expérimentale de techniques d'estimation. De plus, il regroupe les trois domaines d'application de COSMIC-FFP définis dans le chapitre précédent :

- logiciels de gestion des affaires que l'on retrouve typiquement dans les banques et assurances
- logiciels en temps-réel dont la tâche est de garder le contrôle des événements en temps réel
- logiciels hybrides (gestion & temps-réel)

La brève description de ces cas ci-dessous confirme l'hétérogénéité de l'échantillon :

- **Automatic Line Switching** [6] : système en temps-réel contrôlant la qualité d'une ligne de communication. Si celle-ci se dégrade ou défaille, une ligne de secours prend automatiquement le relais.
- **SAVAT Gateway (SAGA) System** [11] : système d'e-business en temps-réel permettant la gestion d'achats en ligne de produits SAVAT.
- **C-Registration System** [9] : système de gestion procurant aux étudiants la possibilité de s'inscrire pour les cours. Les professeurs y renseignent leurs cours et enregistrent les notes attribuées aux étudiants.
- **Collegiate Sports Paging System** [10] : système payant de gestion en ligne permettant aux abonnés de suivre les événements sportifs souhaités. Ils reçoivent ainsi toutes les informations relatives à ceux-ci dès leur parution.
- **Hotel Accommodation System (Reservation)** [4] : système de gestion de réservation de chambres d'un hôtel.
- **L-Euchre card game system** [7] : jeu de cartes en ligne et temps-réel.
- **Rice Cooker System** [8] : système en temps-réel contrôlant un cuiseur de riz.
- **Valve Control System** [5] : système en temps-réel contrôlant le comportement d'une valve attachée au système de changement de vitesses automatique d'une voiture.

Degrés d'avancement de l'analyse dans la phase de mesure

L'objectif poursuivi se résume donc à la recherche d'un quelconque moyen formel (équation, arbre de décision, etc) permettant d'estimer à un taux raisonnable d'erreurs la taille fonctionnelle d'une application et ce, sur base d'informations partielles de l'analyse de celle-ci. Afin d'atteindre cet objectif, d'autres problèmes sous-jacents devront au préalable être traités. Il est évident que plus l'information disponible sur l'analyse de l'application est importante, plus la mesure sera précise. L'une des priorités consiste donc à trouver le seuil d'information nécessaire à une estimation de qualité acceptable.

Le degré d'avancement de l'analyse fonctionnelle d'un projet peut être, comme l'ont souligné

Abran et Azzouz [22], divisé en trois niveaux comme représenté à la figure 3.2. Une mesure est également associée à chaque niveau.



FIG. 3.2 – Différents niveaux d'abstraction de la phase de développement

Une courte explication est donnée ci-dessous, les niveaux étant classés par ordre d'avancement (du moins au plus avancés) dans le cycle de développement du logiciel :

- Le niveau 1 correspond aux phases de modélisation et de spécification des exigences, la taille fonctionnelle est évaluée à l'aide des diagrammes de cas d'utilisation (UC). L'unité à ce niveau est identifiée comme Ufsu pour *Use Case functional size unit*. Cette taille se calcule en prenant la somme pour chaque UC du nombre d'acteurs et d'interactions² (i.e. pour n use cases, on aura : $Taille(Cfsu) = \sum_{i=0}^n (\#Acteur_i + \#interaction_i)$).
- Le niveau 2 correspond à la phase d'analyse, la taille fonctionnelle y est évaluée à partir des scénarii qui réalisent les UC précédents. L'unité à ce niveau est identifiée comme Sfsu *Scenario functional size unit*. Cette taille se calcule en prenant la somme arithmétique des scénarii suivi du nombre de groupes de données distincts³
- Le niveau 3 correspond à la phase d'analyse approfondie, la taille fonctionnelle y est évaluée à partir du détail des scénarii. L'unité à ce niveau est identifiée comme Cfsu⁴ *COSMIC functional size unit*. Il s'agit donc ici de calculer le nombre d'entrées, sorties, lectures et écritures des groupes de données.

Une première étape poursuivie vise à rechercher les données (par use case : #acteurs, #interactions, #groupe de données, #scénarii, taille fonctionnelle COSMIC (CFSU)) des huit cas

²Dans le langage UML, une interaction représente l'interaction d'un acteur avec un UC, c'est-à-dire le lien tracé entre les deux

³même définition que dans ISO 19761[3] : Ensemble distinct non vide, non ordonné et non redondant de types d'attributs. Chaque type d'attribut dans un groupe de données décrit un aspect complémentaire du même objet d'intérêt. Un groupe de données est caractérisé par sa persistance

⁴défini dans la norme ISO 19761[3]

3.3 - Recherche d'une nouvelle technique

décrits ci-dessus. Ces données sont représentées dans le tableau 3.2.

Use-Case	Acteurs	Interaction	Scénario	Groupe de données	Unité Cosmic (Cfsu)
set target temperature	1	2	3	1	5
control heater	2	2	3	1	4
control indicator lamp	1	1	3	1	3
restore backup line	1	1	4	1	5
restore work line	1	1	4	1	5
remove backup line	1	1	4	1	5
remove work line	1	1	4	1	5
conditional switch to work line	1	1	4	1	5
conditional switch to backup line	1	1	5	1	5
forced switch to work line	1	1	3	1	3
forced switch to backup line	1	1	3	1	3
work line failed	1	1	3	1	5
backup line failed	1	1	3	1	5
work line degraded	1	1	3	1	5
backup line degraded	1	1	4	1	5
work line cleared	1	1	3	1	5
backup line cleared	1	1	3	1	5
subscribe	1	1	4	2	13
provide feedback	2	2	2	2	8
edit profile	1	1	4	2	13
read content on website	1	1	6	3	14
pay fee with credit card	1	1	4	2	12
post advertising content	1	1	7	3	17
print advertiser reports	1	1	4	3	12
send content	1	1	3	1	5
send page	2	2	3	1	4
approve story	1	1	10	8	38
create reservation	1	1	6	1	8
update reservation	1	1	5	1	12

TAB. 3.2: (suite)

Use-Case	Acteurs	Interaction	Scénario	Groupe de données	Unité Cosmic (Cfsu)
accept reservation	1	1	4	1	8
confirm reservation	1	1	5	1	10
select reservation	1	1	4	1	8
cancel reservation	1	1	5	1	12
room type report	1	1	3	1	8
hydraulic valve	2	2	10	1	12
host a game	1	1	3	1	6
new game	2	2	4	1	5
going alone	2	2	2	1	4
choosing trump	2	2	2	1	5
playing a hand	2	2	2	1	3
choose a seat	2	2	1	1	4
dealing of cards	2	2	1	1	3
joining a game	1	1	4	1	5
error message	1	1	2	1	3
ending a hand	1	1	2	1	3
scoring a hand	1	1	5	1	5
winning a trick	1	1	3	1	4
winning/loosing a game	1	1	3	1	5
lost IP connection	1	1	1	1	4
keeping a score	1	1	2	1	3
view report card	1	1	3	1	5
register for courses	2	2	5	5	30
select courses to teach	2	2	4	1	8
submit grades	1	1	3	1	6
maintain professor infor- mation	1	1	2	4	16
maintain student infor- mation	1	1	2	4	16
close registration	2	2	5	2	12
logon	1	1	3	1	3

TAB. 3.2: (suite)

Use-Case	Acteurs	Interaction	Scénario	Groupe de données	Unité Cosmic (Cfsu)
get account balance	1	1	2	1	4
delete product category	1	1	6	1	12
get product categ serial	1	1	2	1	4
logon	1	2	4	1	7
product categ list	1	2	2	1	4
change pwd	1	1	5	1	11
order account statement	1	1	5	1	7
register product category	1	1	5	1	8
logoff	1	1	4	1	6
order goods	1	1	5	1	7
freeze logon id	1	1	3	1	4
users product category list	1	1	3	1	5
adhoc catalog order	1	1	2	1	3
display statement summary	1	1	4	1	9
get account serial	1	2	2	1	4
list orders for account	1	3	4	1	10
get last nr transaction	1	1	2	1	4

TAB. 3.2: Echantillon d'analyse

L'accent est ensuite mis sur la recherche d'un modèle général (équation, arbre de décision, etc) prenant en entrée les données des premier et second niveaux et produisant en sortie une estimation des données du dernier niveau comme illustré à la figure 3.3. Afin de pouvoir être accepté, ce modèle devrait atteindre un certain degré de précision dans sa prédiction tout en permettant d'avoir une idée de l'erreur commise.

Des techniques de statistique et de data-mining conviennent parfaitement comme outils d'analyse des données du tableau 3.2 au vu des exigences demandées. Ces techniques vont dès lors être mises en oeuvre dans les prochaines sections de ce chapitre afin de tenter d'obtenir un modèle acceptable pouvant conduire à une estimation de la taille fonctionnelle tôt dans sa

phase de développement.

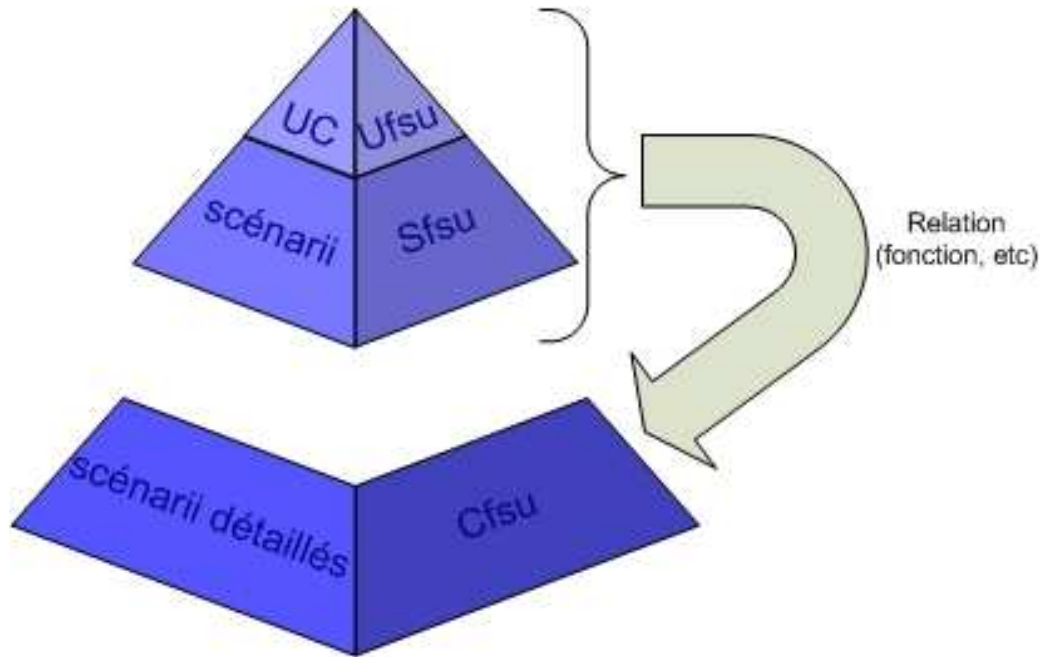


FIG. 3.3 – Modélisation du processus à atteindre

3.3.2 Outils et techniques

Un plan d'analyse a été mis au point afin de traiter les données ci-dessus par un large panel de techniques de data-mining ; ceci afin de sélectionner celle qui offre les meilleurs résultats. La première analyse présentée est celle qui a fourni les meilleurs résultats en termes de prédictivité. Il s'agit de la méthode de régression linéaire multiple. Une attention particulière lui sera accordée par rapport aux autres techniques. Ces dernières seront ensuite regroupées par type de données qu'elles prennent en entrée. La première série prendra comme échantillon des données quantitatives tandis que pour la seconde série, la variable aléatoire Cfsu sera rendue qualitative. Toutes les techniques utilisées sont des techniques de classification supervisée⁵.

Le logiciel de data-mining utilisé pour les diverses analyses qui vont suivre se nomme *Weka*[48] (Waikato Environment for Knowledge Analysis). Weka est un logiciel développé par les chercheurs de l'université de Waikato en Nouvelle Zélande. Il regroupe un ensemble d'algorithmes d'apprentissage, de reconnaissance des formes et de data mining recouvrant les méthodes de classification supervisées et non supervisées. Une structure de tableau de données

⁵supervisé signifie que les résultats devant être obtenus sont fournis afin de faire converger les données en entrées vers ceux-ci.

normalisé (le format ARFF) et des outils d'interrogation de bases de données permettent d'avoir une entrée unique. Actuellement WEKA contient des modules de pré-analyse et de visualisation, de classification, de régression et des algorithmes construisant des règles d'association. WEKA est un « open source software » sous la licence publique GNU.

3.4 Technique de régression linéaire multiple

3.4.1 Le modèle généralisé

La première technique à envisager dans ce contexte est celle de la régression linéaire multiple. Cet outil statistique très efficace et relativement intuitif semble être un très bon choix pour une solution linéaire, c'est-à-dire sous forme d'équation du premier degré en ses variables explicatives⁶. La régression multiple représente une extension de la notion de régression simple. Elle permet de prendre en compte plusieurs variables explicatives et rend compte de la linéarité ou non des données.

Le but de cette technique est donc d'exprimer une dépendance de y comme fonction de plusieurs variables explicatives x_i , de manière à prévoir la valeur de y connaissant celle des x_i .

$$y = f(x_1, \dots, x_n) \quad \text{pour } n \text{ variables explicatives}$$

Le modèle généralisé de régression linéaire multiple pour n variables explicatives prend la forme suivante :

$$E(y) = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n$$

avec y : la variable expliquée,

$E(y)$: l'espérance de y ,

$x_{1\dots n}$: les n variables explicatives

$\beta_{0\dots n}$: les coefficients de régression

⁶Une variable est dite explicative x si elle influence une autre variable, dite expliquée y c'est-à-dire qui fait l'objet de l'étude statistique.

3.4.2 Transposition du modèle généralisé

Soient y : l'estimation de la taille recherchée (Cfsu),

$E(y)$: l'espérance de y ,

$x_{1...4}$: les variables explicatives du tableau de données 3.2, La transposition du
soient les UC, interactions, scenarii et les groupes de données

$\beta_{0...4}$: les coefficients de régression à trouver

modèle généralisé aux données qui nous concernent donne l'équation suivante :

$$E(y) = \beta_0 + \beta_1 x_1 + \cdots + \beta_4 x_4$$

L'utilisation de cette technique va permettre de répondre aux questions de linéarité ou non des données du tableau 3.2 et, dans l'affirmative, donnera la relation de dépendance linéaire entre la variable recherchée (y) et les quatre variables explicatives ($x_{1...4}$). Ainsi, les inconnues sont les cinq coefficients β de l'équation.

3.4.3 L'hypothèse H_0

Les tests statistiques fonctionnent tous sur le même principe qui consiste à énoncer une hypothèse devant être applicable sur l'ensemble des données et pas seulement sur l'échantillon. Il suffit alors de vérifier si les observations sont vraisemblables dans le cadre de cette hypothèse pour s'assurer que l'échantillon est bien représentatif de l'ensemble des données.

L'hypothèse à tester est appelée H_0 connue aussi sous le nom d'hypothèse nulle [49]. Elle s'accompagne de son hypothèse alternative H_1 . Le test a pour but de valider ou rejeter H_0 (et par conséquent à tirer la conclusion inverse pour H_1). En général, on peut qualifier de peu vraisemblable — i.e. pouvant être rejetée — toute hypothèse située hors de l'intervalle de confiance. Par contre, on peut admettre comme plausible ou acceptable, toute hypothèse située dans l'intervalle de confiance défini. Si le résultat du test amène à accepter l'hypothèse nulle H_0 , nous pouvons en déduire qu'il n'y a rien à conclure à partir des observations concernées. En revanche, le rejet d' H_0 signifie que la répartition des réponses recèle des informations particulières qui ne semblent pas être dues au hasard et qu'il convient d'approfondir.

Dans le cas de la régression linéaire⁷, on fait l'hypothèse d'indépendance entre la variable Cfsu et les autres. On a donc

$$H_0 : \beta_{1...4} = 0$$

⁷cas spécifique à la régression linéaire car dans d'autres situations, l'hypothèse pourrait être par exemple une moyenne.

C'est ainsi que pour chaque β , l'hypothèse H_0 est vérifiée en prenant un intervalle de confiance de 95%⁸ ; en d'autres termes, l'échantillon a dix-neuf chances sur vingt d'être représentatif de l'ensemble des données.

Afin de pouvoir calculer un intervalle de confiance, il faut connaître la distribution de l'échantillonnage. Le théorème central limite⁹ permet d'affirmer que la distribution est **Normale** puisque la taille de l'échantillon du tableau 3.2 est de 72 et que pour satisfaire au théorème central limite, n doit être supérieur ou égale à 20. La figure 3.4 illustre un exemple d'un intervalle de confiance avec une distribution normale. L'acceptation ou le rejet de l'hypothèse nulle en régression est traité comme suit :

Si la valeur zéro de l'hypothèse nulle n'est pas contenue dans cet intervalle, on peut alors la rejeter et conclure la dépendance du paramètre associé au coefficient β avec la variable expliquée (valeur Cfsu dans ce cas).

Dans l'exemple de la figure 3.4, la valeur zéro fait partie de l'intervalle impliquant le non-rejet d' H_0 . Cela signifie que la variable testée ($x_{1...4}$) est indépendante de la variable cible (Cfsu) avec 95% de chance que l'échantillon soit représentatif des données.

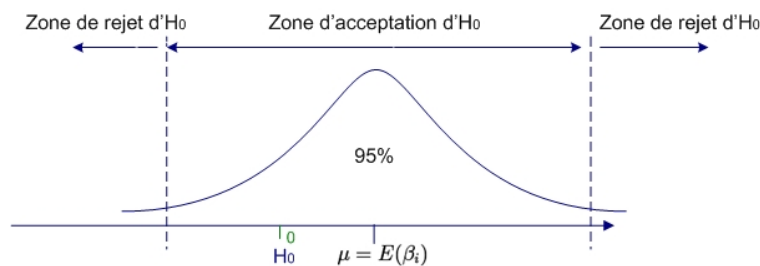


FIG. 3.4 – Zone d'acceptation d' H_0

Le test de l'hypothèse nulle dans le cadre de la régression linéaire est réalisé sur base des résultats d'une première analyse par régression des données¹⁰. La section suivante fait part de ces résultats.

⁸95% étant le seuil généralement choisi.

⁹Soit X_1, X_2, \dots, X_n un échantillon aléatoire de taille n extrait d'une population de moyenne μ et de variance σ^2 . Alors la distribution d'échantillonnage de \bar{X} est approximativement (lorsque n est grand), une distribution Normale de moyenne μ et de variance $\frac{\sigma^2}{n}$.

¹⁰le logiciel utilisé s'intitule Weka[48]

3.4.4 Analyse de régression non épurée des variables exogènes

La première étape pour obtenir un modèle de regression linéaire consiste à épurar les variables exogènes¹¹ éventuelles pouvant le biaiser. A cette fin, il faut calculer l'intervalle de confiance pour chaque variable explicative ($x_{1...4}$) et vérifier si elles rejettent bien l'hypothèse nulle.

Le tableau 3.3 illustre les résultats fournis par la première analyse par régression linéaire sur les données du tableau 3.2.

Variable explicative	Coefficient	écart type du coefficient
constante(β_0)	-2,9923	0,8969
acteurs(β_1)	-0,5583	0,846967
interactions(β_2)	1,0614	0,6906
groupes de données(β_3)	1,1909	0,1445
scénario(β_4)	3,9694	0,2165

TAB. 3.3 – Résultat de la régression.

L'intervalle de confiance à 95% se calcule de la façon suivante :

$$\beta \pm t_{0,975}SE$$

Où β est le coefficient de la variable explicative,

SE est son écart type et

$t_{0,975}$ est le quantile d'ordre $1 - \frac{\alpha}{2}$ ¹²de la loi de Student à $(n - 1)$ degrés de liberté ¹³.

L'examen du tableau 3.4 ne permet pas de rejeter l'hypothèse nulle pour la variable acteur et interaction. En effet, leur intervalle de confiance inclut tous deux la valeur zéro, signe d'indépendance (H_0). La figure 3.4 illustre le cas de ces deux variables.

Une autre manière de le prouver consiste à calculer *la p-value*. Celle-ci résume très clairement le degré de concordance entre les données et (H_0). L'avantage de la p-value est qu'elle fournit une idée très précise du risque d'erreur qu'on encourt en rejetant H_0 alors que H_0 est

¹¹une variable exogène est une variable qui n'apporte pas de valeur ajoutée (à l'équation de régression dans ce cas).

¹² α représente l'erreur de l'intervalle de confiance (ici, $1 - 0,95 = 0,05$). Il faut le divisé en deux puisque il est réparti à gauche et à droite de l'intervalle.

¹³La population de l'échantillon (le nombre de UC dans ce cas)

Variable explicative	Intervalle de confiance	p-value	Rejet d' H_0
acteurs	$-1,58 \leq E(\beta_1) \leq 0,47$	0,5119	NON
interactions	$-0,32 \leq E(\beta_2) \leq 2,44$	0,129	NON
groupes de données	$0,9 \leq E(\beta_3) \leq 1,48$	0	OUI
scénario	$3,54 \leq E(\beta_4) \leq 4,4$	0	OUI

TAB. 3.4 – Tableau de rejet de l’hypothèse nulle

vraie. Dans ce cas-ci, on rejette donc H_0 au niveau de signification $\alpha = 0,05$ ¹⁴. La variable acteur a une p-value de 0,512, ce qui est supérieur à 0,05, donc on ne peut pas rejeter H_0 . Il en va de même pour la variable interaction. Ces deux variables sont donc exogènes au modèle et faussent ce dernier.

Par contre, les variables groupe de données et scénario ont une p-value de 0 ($0 < 0,05$) donc on rejette H_0 . Autrement dit, le risque d’erreur que l’on commet, c’est-à-dire le risque d’avoir pris une décision fausse, est égal à 0.

3.4.5 Analyse de régression épurée

Une nouvelle analyse par régression linéaire multiple, après avoir rejeté les 2 variables exogènes, donne les résultats suivant :

Variables	Coefficient	écart type du coefficient	p-value
constante(β_0)	-2,2326	0,5355	0,000084
groupes de données(β_1)	1,1707	0,1446	0
scénario(β_2)	3,9614	0,2169	0

TAB. 3.5 – Résultat de la seconde régression.

Les faibles valeurs de la colonne *p-value* de la table 3.5 attestent du caractère significatif et de la qualité de l’échantillon [35]. Une valeur nulle signifie une projection parfaite de l’échantillon sur la population. Celui-ci est par conséquent très représentatif de la population en général sous l’hypothèse nulle.

¹⁴0,05 car l’intervalle est construit à 95%.

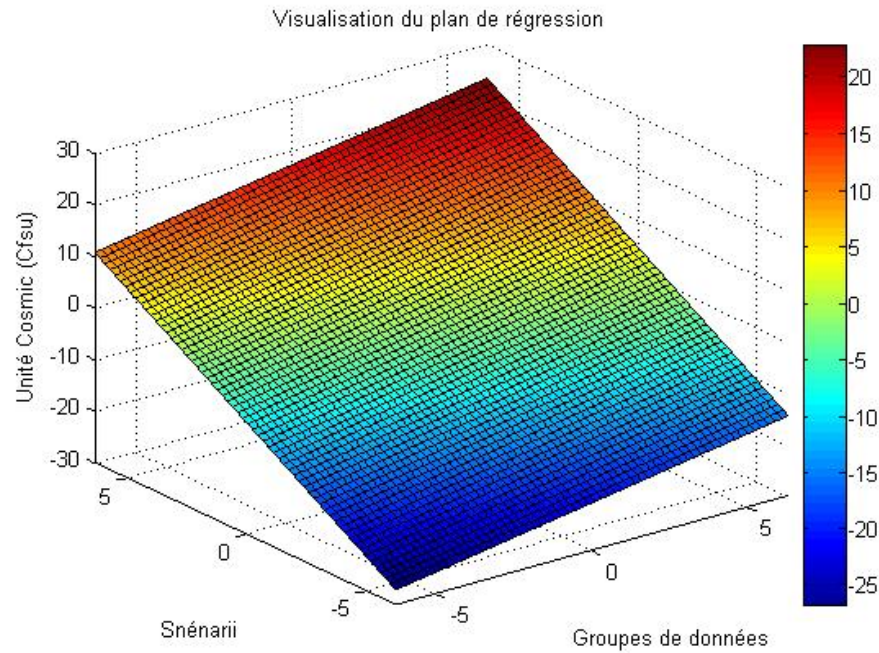


FIG. 3.5 – Visualisation du plan de régression

3.4.6 Solution obtenue de la technique de régression linéaire multiple

L'équation obtenue en solution est la suivante :

$$Y = -2,23 + 1,17X_1 + 3,96X_2$$

Elle peut être visualisée par le plan¹⁵ illustré en 3.5 :

3.4.7 Estimation de l'erreur du modèle

Afin de s'assurer de la qualité de ce nouveau modèle, une estimation de l'erreur va permettre de déterminer formellement la validité de ce modèle. Pour faire une prédiction de la taille d'une application (en Cfsu) sans disposer des données scénarii et groupes de données, la seule façon de procéder aurait été de prendre la moyenne de l'échantillon, soit \bar{y} . L'erreur de prédiction serait alors de $(y - \bar{y})$.

¹⁵Ce plan n'est que la visualisation de l'équation donnée par la régression. Il est évident qu'en réalité, il ne peut y avoir de valeur négative pour la taille d'une fonctionnalité, le nombre de scénarii et le nombre de groupes de données.

Mais, lorsque le nombre de scénarii et de groupes de données sont connus, on sait faire beaucoup mieux : la prédiction de y est effectuée à l'aide de \hat{y} ¹⁶. Ceci permet de réduire considérablement l'erreur puisqu'on explique maintenant une fraction plus importante de l'écart, soit $(\hat{y} - \bar{y})$.

On a que[24] :

$$\sum_{k=1}^n (y_k - \bar{y}_k)^2 = \sum_{k=1}^n (\hat{y}_k - \bar{y}_k)^2 + \sum_{k=1}^n (y_k - \hat{y}_k)^2$$

$$SS_t = SS_e + SS_r$$

avec y_k la valeur observée,

\bar{y}_k la moyenne et

\hat{y}_k la valeur calculée

Ainsi, sur base des données du tableau 3.2 et de la nouvelle équation, l'analyse de la variance va permettre d'obtenir le coefficient de détermination et de corrélation. Cette analyse tient compte de la variance expliquée et résiduelle. La somme de ces deux variances étant la variance totale.

- SS_t est la *somme totale des carrés des écarts* . Elle traduit la dispersion des valeurs observées de y par rapport à la moyenne.
- SS_e est la *somme expliquée des carrés des écarts* . Elle traduit la dispersion des valeurs calculées de y par rapport à la moyenne.

Dans ce modèle, les variances sont illustrées dans la table 3.6. Si l'équation de la droite

Variance	Valeur
expliquée (SS_e)	2177
résiduelle (SS_r)	255
totale (SS_t)	2432

TAB. 3.6 – Variance

représente correctement les valeurs expérimentales, on a :

$$SS_e \rightarrow SS_t \iff r^2 = \frac{SS_e}{SS_t} \rightarrow 1$$

r^2 est appelé *coefficient de détermination*. Il représente la part de la variance totale de Y expliquée par l'ajustement réalisé grâce à la régression. Ici, il est égal à 0,895. Cela signifie que 89,5% de la variabilité est expliquée par les groupes de données et les scénarii. Le reste de la variabilité est dû à des effets (autres variables explicatives) qui n'ont pas été mesurés. Ce résultat est donc excellent et démontre une très grande qualité du modèle.

¹⁶ \hat{y} représente la valeur calculée sur la droite de régression.

En en prenant la racine carrée, on obtient le coefficient de corrélation (0,946).

Cependant, r^2 peut croître légèrement en introduisant un régresseur peu significatif¹⁷. C'est pourquoi il est souvent souhaitable d'effectuer une correction pour tenir compte de ce phénomène en réduisant r^2 de manière appropriée. Dans notre cas, il ne devrait pas y avoir une grosse différence comme les deux variables de régression utilisées sont, comme il l'a été démontré précédemment, très significatives.

Avec k régresseur, le r^2 ajusté est défini comme :

$$\bar{r}^2 \equiv \frac{(n-1)r^2 - k}{n - k - 1}$$

Ce qui nous donne pour résultat un \bar{r}^2 de 0,8921. Les deux régresseurs utilisés sont donc indéniablement significatifs et expliquent même pratiquement 90% de la variabilité totale.

3.4.8 Conclusion

L'équation de régression linéaire obtenue en 3.4.6 estime avec une grande précision la mesure COSMIC d'une application à partir de ses cas d'utilisation. La mesure fonctionnelle de chacun d'eux peut être estimée par régression linéaire multiple depuis la connaissance du nombre de scénarii propres au UC relatif et le nombre de groupes de données différents que celui-ci manipule. La sommation de la mesure COSMIC de chaque UC fournit une estimation valable de la mesure fonctionnelle de l'ensemble de l'application.

La précision des résultats obtenus confère une réelle valeur à cette méthode d'estimation. Un chef de projet désireux de connaître la taille fonctionnelle de son projet devrait pouvoir compter sur une estimation s'il ne dispose que d'un petit budget ou un délai assez bref ou encore par simple souci d'économie de temps et d'argent.

Dans les annexes, la première section reprend un tableau récapitulatif des valeurs réelles et estimées par UC afin de concrétiser la valeur de cette équation de régression.

3.5 Autres techniques

Cette section du chapitre rend compte des autres résultats de recherche obtenus mais ayant fourni un modèle d'une moins bonne qualité que la régression linéaire. Il s'agit de modèles

¹⁷un régresseur peu significatif est un régresseur dont le coefficient de régression est proche de 0.

générés à base d'algorithmes d'arbres de décision, de règles de décision ou encore de réseaux de neurones. Cette section prend toute sa signification en apportant la preuve qu'aucune piste de recherche n'a été négligée et évitera d'éventuels travaux de recherche menés dans un contexte similaire.

Il existe plusieurs algorithmes permettant la construction d'arbres de décision. Cependant, un seul d'entre eux accepte les variables dites *continues*¹⁸. La sous-section suivante traitera de son analyse. Ensuite, les recherches seront élargies sur des modèles donnant comme résultat une variable discrète¹⁹. Dans ce cas, les valeurs à prédire Cfsu seront discrétisées. Afin de comparer la prédictivité de ces modèles entre eux (à variables continues et discrètes), une nouvelle définition de l'erreur sera donnée. Les nouveaux modèles générés pourront dès lors être comparés avec celui de régression linéaire multiple.

Il est à noter que seules les variables *groupe de données* et *scénario* seront utilisées puisque les deux autres ne peuvent que biaiser le modèle.

3.5.1 Technique des réseaux de neurones

Vue d'ensemble

Les réseaux de neurones ont été imaginés sur base de l'activité du cerveau humain. Celui-ci est constitué d'un très grand nombre (\pm dix milliards) de cellules nerveuses qui traitent l'information. Chaque cellule fonctionne comme un simple processeur, ce sont seulement les innombrables interconnexions couplées à un traitement des données en parallèle qui fondent la puissance de notre cerveau.

L'algorithme utilisé par Weka s'appelle « Perceptron Multicouches » (PMC). Il s'agit du plus célèbre des réseaux de neurones supervisés²⁰. Cette célébrité est d'ailleurs justifiée par le fait qu'à ce jour, le PMC est le meilleur régresseur et le meilleur classifieur connu[14].

Il fonctionne sur le même principe qu'un cerveau humain.

Comme le cerveau humain, un réseau de neurones est composé de neurones et de connexions

¹⁸une variable numérique continue appartient aux réels (\mathbb{R})

¹⁹Une variable aléatoire est discrète si elle ne prend qu'un nombre fini ou dénombrable de valeurs.

²⁰Dans ce type d'apprentissage, le réseau s'adapte par comparaison entre le résultat qu'il a calculé, en fonction des entrées fournies, et la réponse attendue en sortie. Ainsi, le réseau va se modifier jusqu'à ce qu'il trouve la bonne sortie, c'est-à-dire celle attendue, correspondant à une entrée donnée.[31]

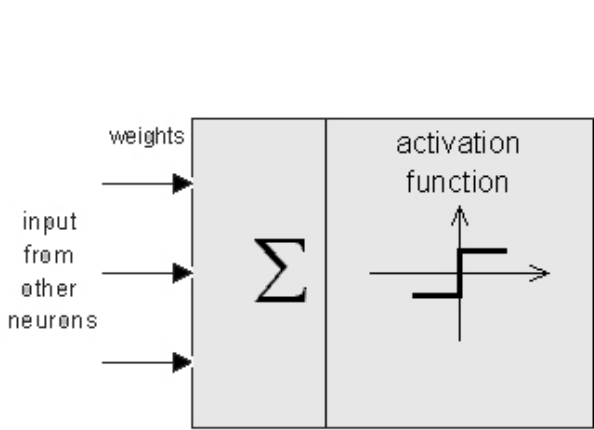


FIG. 3.6 – Structure d'un neurone humain

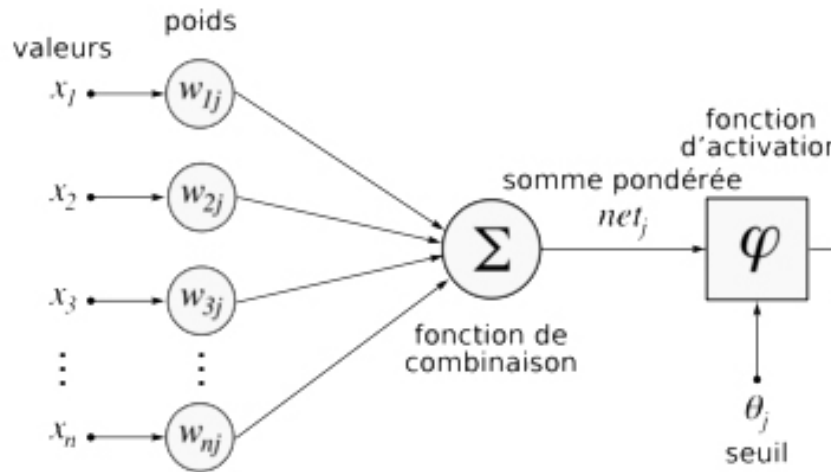


FIG. 3.7 – Structure d'un neurone artificiel[47]

entre eux. Les neurones relayent l'information sur leurs connexions sortantes aux neurones voisins. Ces connexions sont désignées par les poids simulant l'information électrique. Le neurone reçoit l'information par ses différentes entrées. Celle-ci est traitée par une fonction de propagation qui additionne les valeurs de toutes les entrées. En fait, il renvoie le produit scalaire entre le vecteur des entrées et le vecteur des poids synaptiques. La valeur résultante est finalement comparée avec une certaine valeur seuil par la fonction d'activation du neurone. Si la valeur résultante est supérieure à la valeur seuil, le neurone s'active, sinon, il reste en état inactif. La fonction d'activation sert à introduire une non-linéarité dans le fonctionnement du neurone. Les trois fonctions classiques de fonction d'activation[32] sont :

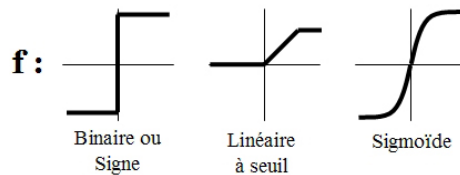


FIG. 3.8 – Les trois types de fonction d'activation

1. la fonction binaire : $f(x) = \pm 1$
2. la fonction linéaire : $f(x) = \frac{1}{1+e^{2x}}$
3. la fonction sigmoïde : $f(x) = ax$

Finalement, si le neurone est activé, il propage son état interne à tous les autres neurones connectés.

Modèle Weka

L'algorithme utilisé par Weka donne les résultats illustrés à la figure 3.9. Ceux-ci sont

```

Linear Node 0
  Inputs  Weights
  Threshold  1.091820467878494
  Node 1  -2.3796918428190605
Sigmoid Node 1
  Inputs  Weights
  Threshold  -0.6678405527190538
  Attrib dataGroups  -0.7386025765244965
  Attrib scenario  -1.7375812488250904
Class
  Input
  Node 0

```

FIG. 3.9 – Modèle du réseau de neurones

représentés visuellement à la figure 3.10 permettant ainsi une meilleure compréhension. La

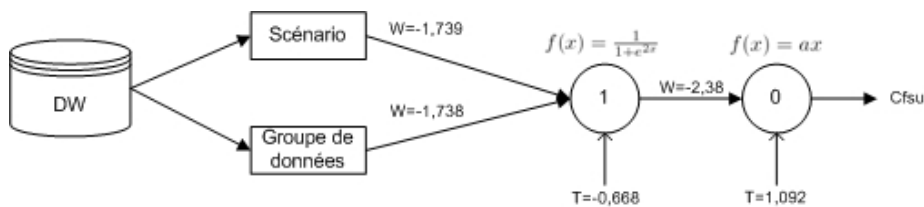


FIG. 3.10 – Représentation du modèle de réseau de neurones

figure 3.10 met en lumière la présence d'une couche cachée composée d'un neurone. Le premier neurone se caractérise par une fonction d'activation sigmoïde tandis que le deuxième par une fonction linéaire.

Le coefficient de corrélation du modèle est de très bonne qualité et est très proche de la régression linéaire. Il s'élève à 0,90.

Critique du modèle

Les réseaux de neurones représentent un puissant outil de prédiction à la fois linéaire et non linéaire par les diverses fonctions d'activation susceptibles d'intervenir dans un réseau. Leur pouvoir de prédiction augmente avec le nombre de noeuds des couches cachées généralement multiples.

Cette technique, bien qu'également efficace dans ce cas de figure, est sous-utilisée vu la relative « simplicité » du modèle recherché. La régression linéaire a en effet déjà prouvé la linéarité du modèle. Cette linéarité est d'ailleurs bien présente dans le noeud de sortie du réseau. Cela

explique par ailleurs la similarité entre les deux coefficients de corrélation.

En conclusion, la linéarité du modèle conduit à rejeter la technique des réseaux de neurones qui complexifie le modèle sans rien apporter en retour. La simplicité et l'efficacité de la régression linéaire supplantent donc pour l'échantillon analysé les réseaux de neurones.

3.5.2 Technique de l'arbre de décision *repTree*

Un arbre de décision est une structure qui permet de déduire un résultat à partir de décisions successives. Pour parcourir un arbre de décision et trouver une solution, il faut partir de la racine puis parcourir une branche jusqu'au dernier noeud appelé feuille. Les noeuds situés entre les feuilles et la racine sont généralement des tests logiques afin de déterminer lequel des deux fils doit être emprunté.

L'algorithme utilisé par Weka pour construire un modèle d'arbre de décision pour une variable continue se nomme « repTree ». Il fournit comme résultat l'arbre suivant :

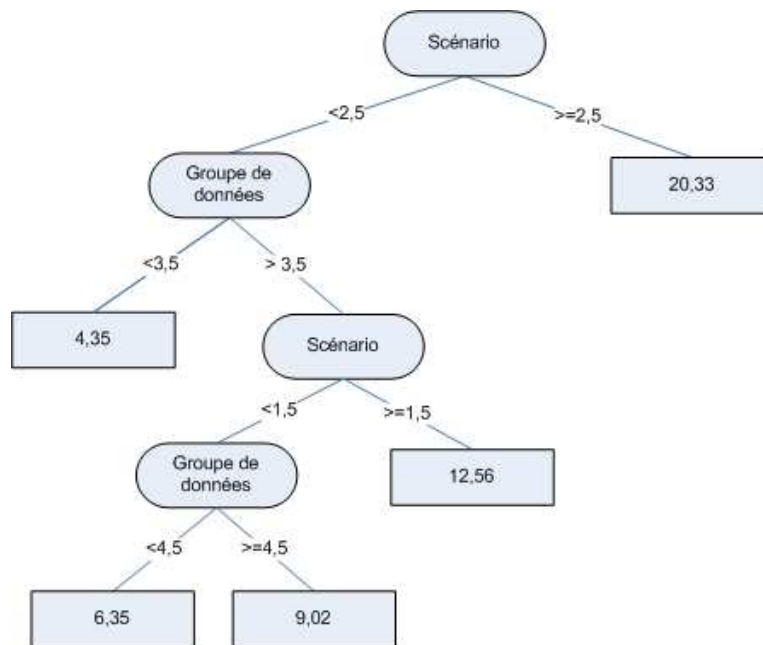


FIG. 3.11 – Arbre de décision

Le coefficient de corrélation de ce modèle s'élève à 0,78. Sa qualité reste effectivement en deçà du modèle par régression linéaire.

Critique du modèle

L'une des grandes différences des arbres de décisions avec le modèle de régression est le nombre de valeurs prédictibles. La régression linéaire n'est pas bornée et s'adapte pour n'importe quelle valeur des variables aléatoires (groupe de données et scénario).

A l'inverse, l'arbre de décision 3.11 opère un genre de classement dans cinq catégories. La valeur de l'estimation de la mesure fonctionnelle d'un UC est donc attribuée selon la catégorie à laquelle il appartient. Un UC ne peut dès lors que recevoir l'une des cinq valeurs déterminées, comprise entre 4Cfsu et 19Cfsu. Cette perte de précision est principalement due à la sensibilité de ce genre de modèle aux écarts de valeurs. Dans cette analyse, deux valeurs se démarquent particulièrement dans les données sources : deux UC ont une taille fonctionnelle de 30 et 38 Cfsu alors que les autres varient entre 3 et 17 Cfsu. Ces écarts influencent le modèle d'une part en augmentant sa complexité et d'autre part en diminuant la précision de son estimation. Sa complexité est en effet moindre sans ces deux valeurs²¹.

Quant à la qualité de précision, le coefficient de corrélation passe de 0,70 à 0,80 une fois ces deux UC enlevés. Cependant, le modèle ne peut alors prédire la taille fonctionnelle d'un UC supérieure à 13,38 Cfsu comme illustré dans la figure 3.12.

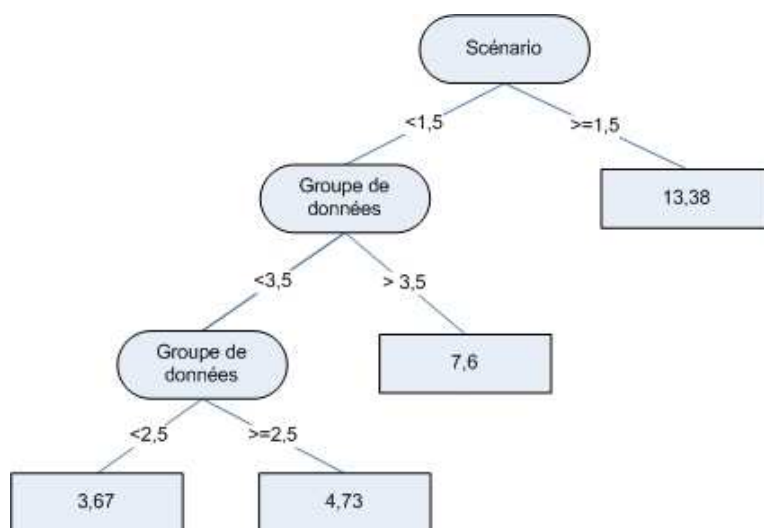


FIG. 3.12 – Arbre de décision élagué

²¹La complexité cyclomatique accroît d'une unité (un test en plus)

3.5.3 Nouvelle définition de l'erreur

Un nouveau moyen d'estimer l'erreur doit être défini afin de pouvoir comparer chacun de ces modèles entre eux. En effet, le coefficient de corrélation ne peut s'appliquer qu'aux modèles continus.

$$e = \frac{1}{n} \sum_{i=1}^n (p_i - r_i)^2$$

avec n : la taille de l'échantillon

p : valeur prédite

r : valeur réelle

On peut observer que cette mesure de l'erreur pénalise fortement les modèles générant de grands écarts entre la valeur prédite et la valeur réelle puisque ces écarts sont élevés au carré. L'interprétation de cette mesure de l'erreur peut se formuler comme suit : plus l'erreur calculée est élevée, moins bonne sera la prédiction du modèle.

Selon ce calcul, le modèle de régression linéaire génère une erreur de $e=3,40$. Celui du réseau de neurones s'élève à $e=5,889$. Quant à l'arbre de décision basé sur l'algorithme repTree, son erreur est de $e=9,97$.

3.5.4 Technique de l'arbre de décision *J48* et *RandomTree*

Weka offre plusieurs algorithmes pour construire un modèle d'arbre de décision pour une variable discrète. L'un se nomme « J48 », l'autre « RandomTree ». Ils génèrent comme résultat l'algorithme suivant :

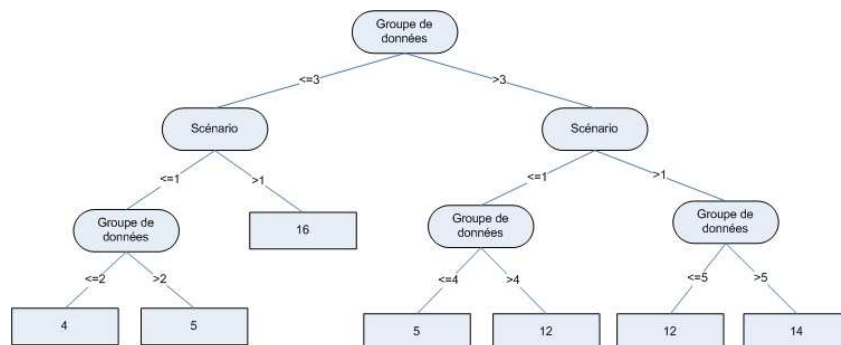


FIG. 3.13 – Arbre de décision J48

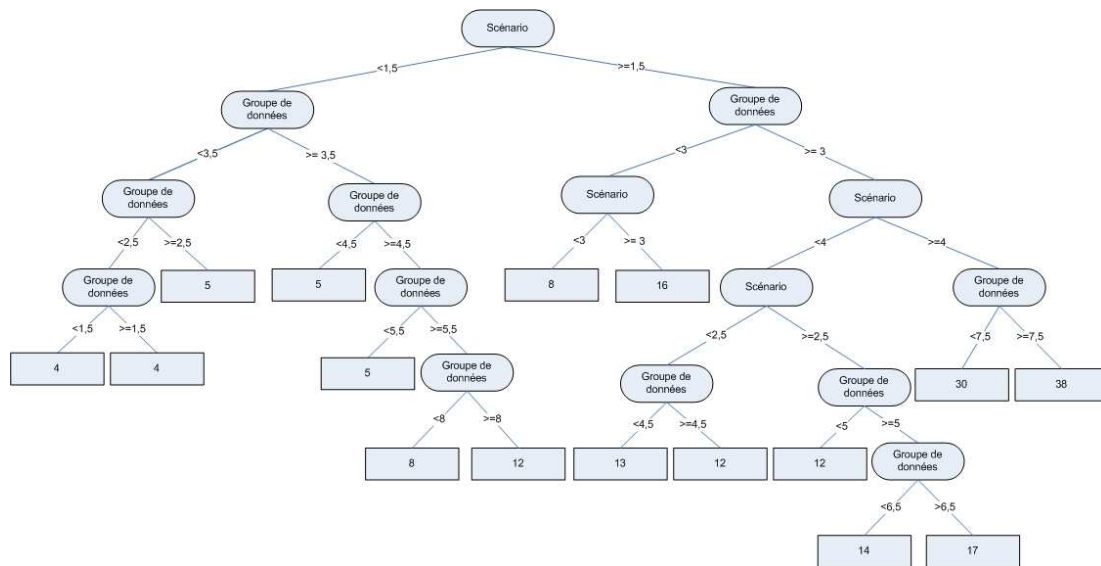


FIG. 3.14 – Algorithme du RandomTree

L'erreur du modèle *arbre de décision J48* illustré ci-dessus s'élève à $e=16,90$.

L'erreur du modèle *random tree* illustré ci-dessous s'élève à $e=4,05$.

Critique du modèle

L'arbre généré par l'algorithme RandomTree est bien plus gros que le précédent. L'algorithme utilisé pour le créer a pour objectif de décrire au maximum l'échantillon et en devient peut-être un peu trop dépendant. Il convient donc de prêter attention lors de l'utilisation de ce modèle. Pour ce qui a trait au premier arbre, il reste assez vague puisqu'il n'offre que cinq feuilles afin de catégoriser.

Un avantage des arbres de décision est leur facilité de compréhension. Par contre, on peut les qualifier d'instables puisqu'une légère variation des données en entrée peut avoir une grosse répercussion dans le choix d'une branche à suivre. L'exemple suivant illustre cette situation, pour l'arbre *RandomTree* :

- si groupes de données ≥ 2 & scénarii $\geq 4 \Rightarrow \text{res} = 30 \rightarrow 38$ Cfsu
- si groupes de données ≥ 2 & scénarii $< 4 \Rightarrow \text{res} = 12 \rightarrow 17$ Cfsu

Autrement dit, pour cet exemple, un scénario en plus fait passer l'estimation de la taille fonctionnelle du UC de l'intervalle $[12-17]$ Cfsu à $[30-38]$ Cfsu ! Ce modèle n'offre pas de valeurs intermédiaires.

3.5.5 Technique Ripple Down Rule Learner rules

D'autres modèles moins connus existent comme le « Ripple Down Rule Learner rules ». Cet algorithme fournit les résultats illustrés par la figure 3.15. Le principe de ce modèle vise à attri-

```
cfsu = 8 (75.0/68.0)
  Except (dataGroups <= 3.5) => cfsu = 5 (25.0/1.0) [14.0/1.0]
    Except (dataGroups <= 2.5) => cfsu = 4 (10.0/1.0) [7.0/0.0]
  Except (scenario > 1.5) => cfsu = 12 (7.0/0.0) [2.0/0.0]
  Except (dataGroups <= 5.5) => cfsu = 12 (15.0/3.0) [9.0/1.0]
    Except (dataGroups <= 3.5) => cfsu = 5 (23.0/0.0) [14.0/0.0]
      Except (dataGroups <= 2.5) => cfsu = 4 (10.0/1.0) [7.0/0.0]
    Except (scenario <= 1.5) and (dataGroups <= 4.5) => cfsu = 3 (10.0/0.0) [2.0/0.0]
      Except (dataGroups > 2.5) and (dataGroups > 3.5) => cfsu = 5 (6.0/0.0) [6.0/0.0]
      Except (dataGroups > 2.5) => cfsu = 5 (14.0/3.0) [6.0/1.0]
```

FIG. 3.15 – Ripple Down Rule Learner rules

buer une valeur par défaut, 8 Cfsu dans ce cas, puis d'énoncer une liste d'exceptions. Pour une entrée donnée, toutes les conditions d'exception sont vérifiées l'une après l'autre. Si une condition est remplie, la valeur de sortie est alors corrigée par la valeur spécifiée dans l'exception. Sinon, la valeur par défaut est attribuée.

Ce modèle n'est pas très intuitif et ne fournit de toute façon pas de très bons résultats puisque l'erreur s'élève à $e=21,96$.

3.5.6 Récapitulatif

Le bref récapitulatif ci-dessous de chacun des modèles rencontrés tout au long de cette analyse permet de se rendre compte de leurs précisions en donnant une représentation de leurs erreurs.

Dans ces six graphes, l'axe des ordonnées désigne la taille fonctionnelle prédite des UC en Cfsu tandis que l'axe des abscisses indique la taille fonctionnelle réelle de ceux-ci. Dans les trois premières figures, comme ce sont des modèles travaillant avec des variables continues, la taille des croix renseigne de l'importance de l'erreur. Plus elle est grosse, plus l'erreur est grande. Dans les trois figures suivantes, comme il n'est pas possible d'évaluer l'importance de l'erreur à travers les modèles qualitatifs, une erreur pour une donnée entrée est signifiée par un carré tandis qu'une croix indique la bonne catégorisation.

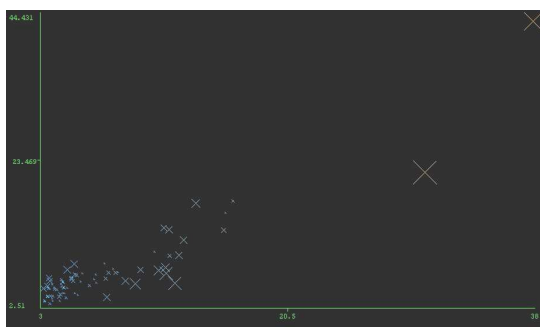


FIG. 3.16 – Erreur : régression

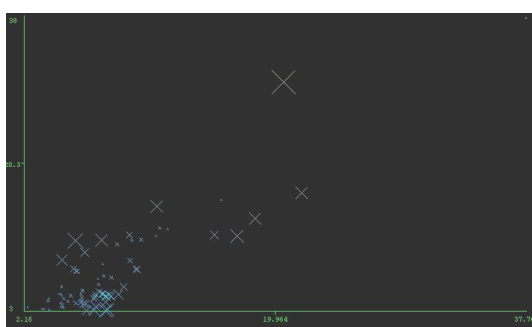


FIG. 3.17 – Erreur : réseau de neurones

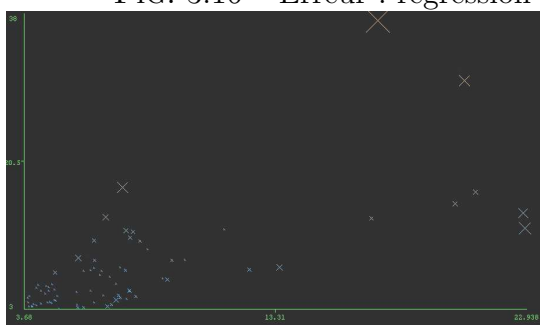


FIG. 3.18 – Erreur : arbre repTree

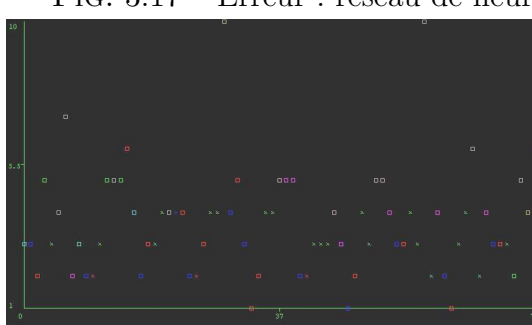


FIG. 3.19 – Erreur : arbre RandomTree

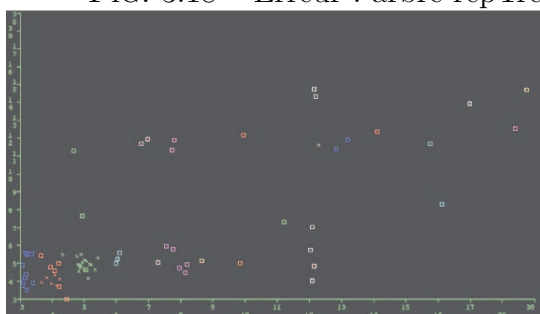


FIG. 3.20 – Erreur : arbre j48

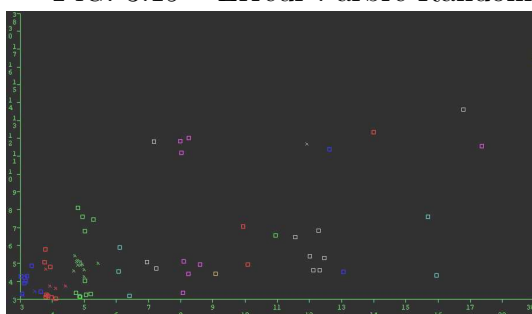


FIG. 3.21 – Erreur : règles RIDOR

légende (Cfsu) : 3 4 5 6 / 8 9 10 11 12 13 14 15 16 17 30 38

On pourrait se demander pourquoi une grosse erreur (croix) dénote sur la figure 3.16 visualisant les erreurs dues à la régression linéaire. Cette erreur provient de l'estimation d'un UC d'une taille de 30 Cfsu. En réalité, seul deux UC ont une taille si élevée. Le modèle de régression a donc été généré principalement sur base de UC de petites et moyennes tailles. Il est donc moins bien adaptée pour les UC de grande taille.

3.6 Conclusion

Tous ces résultats poussent à conclure en la reconnaissance d'un lien de dépendance qu'entretiennent les groupes de données et scénario sur la taille fonctionnelle d'une application mesurée par la méthode COSMIC-FFP.

Il est un fait qu'aucune fonction ou autre modèle ne pourra jamais donner l'estimation de la taille fonctionnelle d'une application avec une certitude sans erreur, sinon, cette mesure n'aurait aucune utilité en soi. Avoir entre les mains un modèle tel que celui de la régression pour ce cas-ci est une réelle aubaine ! Sa qualité d'estimation est excellente au vu de ses différents taux d'erreurs. Cela en fait un outil précieux pour un chef de projet se basant sur la taille fonctionnelle d'une application afin d'en évaluer le coût et le temps de réalisation. S'il ne désire pas être contraint d'effectuer tout le processus de mesure, il pourra obtenir rapidement une estimation valable de la taille fonctionnelle de son application.

Cependant, nonobstant le rejet incontestable de l'hypothèse nulle pour les variables groupes de données et scénario prouvant la représentativité de l'échantillon choisi, le mesureur se doit dans un premier temps d'être prudent avec ces modèles. L'échantillon 3.2 ayant servi de matériel à ces modèles est issu de petites applications²². Bien qu'ils soient variés et évalués par les meilleurs experts, il n'en reste pas moins que ce sont des cas d'école. Il n'est dès lors pas certain que le modèle de régression proposé dans ce chapitre reste en adéquation avec des projets réels d'une taille fonctionnelle pouvant être nettement plus grande.

²²les huit tailles fonctionnelles des cas d'application sont 12 Cfsu(2x), 61 Cfsu(1x), 66 Cfsu(2x), 96 Cfsu(1x), 117 Cfsu(1x) et 136 Cfsu(1x)

Troisième partie

Opérationnalisation de la certification COSMIC-FFP

Chapitre 4

Certification : modèles et approches

Ce chapitre du mémoire consacré aux revues de littérature sur les certifications va permettre de mettre en évidence la partie de l'existant sur laquelle s'est basée le projet de certification de la mesure COSMIC-FFP. Les organisations les plus connues ayant acquis une grande expérience en la matière feront l'objet de cette étude.

Dans un premier temps, afin d'éviter toute confusion et de bien situer la certification par rapport aux autres formes de reconnaissances existantes, une définition précise sera donnée. Ensuite, deux organisations seront analysées et plus spécifiquement leur modèle de certification. Il s'agit d'IFPUG proposant une certification des personnes puis du SEI certifiant les entreprises. Cette analyse permettra d'obtenir à la fin de ce mémoire une idée sur l'état d'avancement et la qualité de la certification de COSMIC-FFP puis du labeur restant à fournir.

4.1 La reconnaissance de l'expertise

Plusieurs types de reconnaissance existent afin de prouver les qualifications d'une personne ou d'une entreprise. Les définitions suivantes permettent de définir clairement les similarités et différences entre les termes les plus usuels.

le certificat :

« Certificates are awarded after you attend a prearranged series of continuing education courses and serve to recognize your successful completion of an educational process. Certificate programs are a good way to build your skills and generally do not require testing or follow-up training[44]. »

la certification :

« Certification is earned after you complete a prearranged series of courses and your knowledge is assessed against a set of industry-relevant standards. Certifications often include on-going requirements that must be met to keep the certification valid. Certification allows you to build your credentials through objective confirmation of your skills[44]. »

l'accréditation :

« Accreditation is a voluntary process to recognise institutions or organisations for meeting standards set by a third party [25]. »

la licence :

« Licensure is a mandatory governmental requirement necessary for individuals to practice in a particular profession [25]. »

Trois types de certification peuvent être distingués :

- La certification de personnes, attestant de leurs compétences professionnelles au regard de critères préétablis.
- La certification de produits ou de services qui atteste des caractéristiques conformes à des spécifications techniques.
- La certification d'entreprise ou d'assurance qualité qui atteste du système qualité conforme à l'un des trois référentiels internationaux ISO 9000 d'une entreprise.

4.2 Modèle de certification de l'IFPUG

4.2.1 Description

L'une des méthodes de mesure de taille logicielle les plus connues à l'heure actuelle est indubitablement celle par point de fonction. Elle est régie par *The International Function Point Users' Group* (IFPUG). Cette organisation à but non-lucratif oeuvre depuis 1986 dans la promotion de sa méthode de mesure dans le domaine du génie logiciel, leur objectif étant leur reconnaissance en tant que leader du marché. [27]

Cette méthode est dérivée des points de fonction d'Albrecht dont la définition initiale est : [20]

« This gives a dimensionless number defined in function points which we have found to be an effective relative measure of function value delivered to our customer. »

Sur cete base, IFPUG s'est orienté sur la taille des fonctionnalités. Elle peut se résumer en ces cinq objectifs : [15]

- Mesurer ce que l'utilisateur demande et reçoit
- Mesurer indépendamment des technologies utilisées pour l'implémentation
- Fournir une mesure de la taille aidant à l'analyse de la qualité et de la productivité
- Fournir un moyen d'estimation du logiciel
- Fournir un facteur de normalisation pour la comparaison des logiciels

4.2.2 Structure organisationnelle du groupe

Afin de mener au mieux leur objectif, cette organisation est divisée en plusieurs « comités ». Cette division de l'organisation décentralise le pouvoir réparti dans les différents secteurs spécialisés.

On retrouve tout de même en haut de l'organigramme un *comité directeur* qui fait fonction de sommet stratégique [34]. Sa fonction est de faire en sorte que l'organisation remplisse sa mission de façon efficace et qu'elle serve les besoins de ceux qui la contrôlent. Le comité directeur exerce des fonctions de supervision directe : allocation des ressources, règlement des conflits, contrôle, diffusion de l'information, leadership. Il est également chargé des relations de l'organisation et de son environnement (contacts à haut niveau, négociation, etc.). Il doit finalement développer la stratégie de l'organisation. Ce comité chapeaute onze comités ayant chacun des tâches bien spécifiques. Quatre d'entre eux ont un lien direct ou indirect avec le processus de certification IFPUG :

- Le *Communications & Marketing Committee* s'occupe de la communication entre l'organisation et ses membres ainsi que de la promotion de leur discipline et de leurs services notamment la certification.
- Le *Counting Practices Committee* tient à jour le manuel de mesure des points de fonction et répond continuellement aux problèmes posés sur un forum. Son rôle est très important dans le processus de certification puisqu'il crée les questions relatives à chaque épreuves d'examination. Il porte la responsabilité de tout le matériel y étant relatif.
- L'*Education Committee* gère tout ce qui a trait aux cours et ateliers notamment en vue de préparer l'épreuve de certification. C'est donc le corps enseignant.
- Le *Certification Committee* est le comité central du processus de certification. Il a pour mission de développer et maintenir un programme professionnel de certification des points de fonction. C'est à lui qu'est confié toute l'organisation administrative d'une session d'examination.

4.2.3 Procédure de certification

L'examen de certification proposé est connu sous le nom de « *Certified Function Point Specialist (CFPS)* ». Il a pour but de reconnaître la connaissance et les compétences des utilisateurs des points de fonction. Cette reconnaissance peut être un atout voir une obligation pour postuler dans certaines compagnies, ce qui lui confère une certaine valeur industrielle.

L'administration

L'administration d'une épreuve CPFS est assez lourde. Un grand nombre de règles la régit tant au niveau du lieu, de la date ou des surveillants d'un examen. Un document intitulé *IFPUG Regional CFPS Examination Rules and Responsibilities* [23] reprend notamment toutes les procédures à suivre afin d'organiser une session.

Les pré-requis

Cet examen ne requiert aucune compétence, certification ou niveau d'étude préalable. La certification CPFS est donc accessible à tous, du moment que les frais d'inscriptions sont acquittés.

Le contenu de l'examen

L'examen, d'une durée de trois heures, est divisé en trois parties :

- la partie de définitions : il s'agit de cinquante questions à choix multiples destinées à tester la connaissance pure des définitions et règles selon le *Counting Practices Manual (CPM)*
- la partie implémentation : à nouveau cinquante questions à choix multiples portant sur une mise en contexte de la théorie.
- la partie cas d'étude : elle vise à évaluer la capacité d'application des règles dans des cas biens réels. L'épreuve compte deux à trois cas d'études.

Le déroulement

Le déroulement d'un examen est lui aussi bien déterminé [23]. Seules les personnes ayant complété le formulaire d'inscription et payé le montant dû trois semaines précédant l'examen pourront participer sur preuve d'une pièce d'identité.

Le surveillant désigné par le comité de certification pour contrôler un examen doit être porteur de sa certification valable au moins encore un an. Il se doit de signer le *Proctor and Witnesses Rules and Responsibilities* pour le responsabiliser sur sa tâche. Il aura reçu avant

l'épreuve un colis scellé contenant les épreuves de l'examen. Celles-ci pourront être ouvertes en la présence de deux témoins directs à l'administration de l'épreuve.

Chaque candidat a droit d'avoir en sa possession le *Counting Practices Manual (CPM)* en version reliée.

L'épreuve dure trois heures et est généralement en anglais. Les candidats non anglophones ont droit à trente minutes supplémentaires pour pallier leur difficulté de la langue. Une fois terminé, le surveillant place immédiatement les copies sous enveloppes scellées à nouveau en la présence de deux témoins.

La correction et remise des résultats

Les examens sont corrigés par le président et le vice-président du comité de certification. Ils délibèrent ensuite sur base des résultats obtenus. La moyenne pour réussir doit être supérieure ou égale à 90% ainsi que minimum 80% dans chacune des parties.

Les résultats sont transmis individuellement par courrier avec en cas de réussite le certificat CFPS. Sinon, une lettre donnant les pourcentages de chaque section les informe des lacunes restant à combler lors d'une éventuelle prochaine tentative.

La publication des lauréats se fait de quatre à six semaines plus tard.

Aucune correction n'est jamais proposée puisque les questions ayant été posées sont susceptibles de revenir ultérieurement.

L'extension

La certification CFPS est valable pendant une période de trois ans. Ceci par souci de maintenir les personnes certifiées CFPS constamment à niveau. Dès lors, s'il désirent rester certifiés, ils peuvent soit recommencer le même examen sus-mentionné valable trois ans, soit adhérer au programme d'extension de certification IFPUG.

Ce programme est particulier dans le sens où aucun examen n'est à présenter. Les candidats doivent remettre un dossier sur leur participation à certaines activités pour un montant total de cent crédits. Ces activités sont généralement des ateliers (*workshop*) auxquels il faut participer. D'autres alternatives sont possibles telles qu'une présentation ou encore la rédaction d'une publication, la prestation d'un cours, etc. Chacune de ces activités octroient au prestataire un quota de crédits. Ces quotas sont cumulatifs sur une période de trois ans et, une fois atteint un

total de cent unités, permettent d'obtenir une prolongation de deux années de leur certification.

Ce programme d'extension peut s'appliquer deux fois successivement après quoi ils se devront de repasser l'examen. Les crédits accumulés sur l'année précédant une première extension peuvent donc faire partie du dossier de la seconde extension.

4.2.4 Critique

L'analyse de cette organisation démontre un grand professionnalisme. Tout d'abord la division en comités indépendants assure une grande qualité à chaque tâche. De plus, cette division confère une grande confiance en l'épreuve puisque personne ne peut connaître tout son parcours. L'analogie de la séparation des pouvoirs au sein d'un pays démocratique l'illustre parfaitement. C'est afin d'éviter tout dérapage que le pouvoir est divisé en trois secteurs : législatif, exécutif et juridique.

Les cours sont organisés et dispensés uniquement par l'*Education Committee*. Indépendamment de celui-ci, le *Counting Practices Committee* réalise avant chaque épreuve les questionnaires. Il est donc le seul à en connaître le contenu puisqu'une fois réalisés, les examens sont mis sous colis scellé. Ce colis est expédié avant l'épreuve au surveillant désigné qui devra à nouveau le sceller une fois l'épreuve terminée. Finalement, la correction est effectuée par le président et le vice-président du *Certification Committee*. Ceux-ci ne peuvent divulguer les corrigés par souci d'égalités des chances lors de futures épreuves.

Par ailleurs, offrir la possibilité d'une extension de deux ans par diverses activités intrinsèquement liées au calcul de points de fonction est une bonne alternative à l'examen, le but recherché étant un maintien continu du niveau de connaissances.

Par ce procédé, IFPUG est certain de conférer une très haute qualité à sa certification et donc aux personnes certifiées CPFS. Les entreprises peuvent dès lors y vouer une grande confiance.

4.3 Modèle de la certification du SEI

4.3.1 Description

La seconde institution analysée est connue depuis plus de 20 ans sous l'abréviation SEI pour *Software Engineering Institute*. Elle est sponsorisée par le département de la Défense américaine et dirigée par l'université du Carnegie Mellon. Cet établissement a pour objectif

d'amender l'utilisation de l'ingénierie logicielle de l'état et tient lieu de ressource nationale en technologie et ingénierie logicielle. [43]

4.3.2 Le modèle CMMI

Le SEI est mondialement connu pour son modèle CMMI. Il s'agit d'un modèle d'évaluation de la capacité en terme de développement qui est utilisé afin d'améliorer la façon de manager, de développer et de maintenir des applications, des équipements et des systèmes.

Issu de l'expérience, le CMMI est un modèle décrivant quelles sont les pratiques à mettre en oeuvre dans une organisation afin d'améliorer et d'assurer la maîtrise des coûts, des délais et des performances des applications, des équipements et des systèmes développés.

Le modèle CMMI est constitué de cinq niveaux de maturité :

- Niveau 1 - Initial : le processus de développement n'est pas défini. La réussite des projets dépend du savoir-faire de quelques personnes clés dans l'organisation. Ce savoir-faire n'est pas formalisé ni partagé par les différentes équipes de projet de l'organisation.
- Niveau 2 - Reproductible : les entrées et les sorties des différentes activités sont gérées et contrôlées. Les règles sont connues et appliquées par les équipes. Le développement est planifié et suivi. Les produits sont vérifiés par rapport aux exigences initiales.
- Niveau 3 - Défini : les meilleures pratiques mises en oeuvre au niveau 2 sur les projets sont généralisées à toute l'organisation. Un projet instancie le processus standard afin de définir la meilleure solution répondant à ses propres objectifs. L'efficacité de chacun des processus est mesurée et renforcée.
- Niveau 4 - Maîtrisé : chaque processus est systématiquement mesuré. Les données sont consolidées et exploitées pour la prise de décision et l'anticipation des risques.
- Niveau 5 - Optimisation : les processus sont totalement maîtrisés et optimisés en permanence. Toute évolution est anticipée et gérée.

La figure 4.1 illustre ces différents paliers associés à leur proportion de risque et de performance.

Le type de certification est donc différent par rapport à IFPUG. Il s'agit ici de certifier la qualité des entreprises et non plus des personnes.

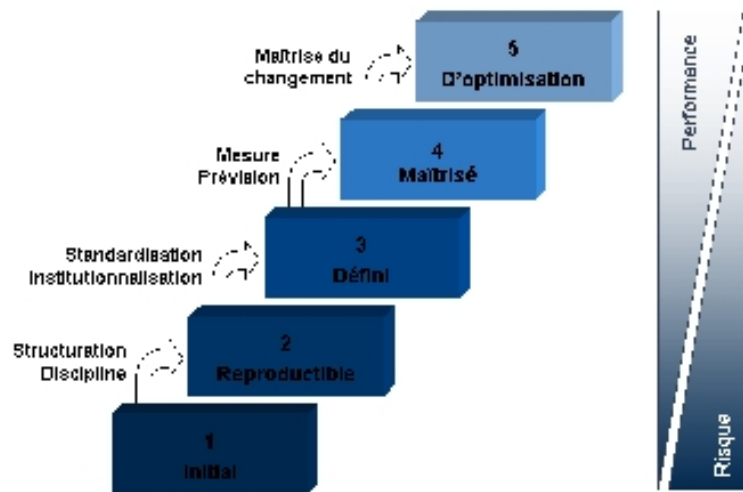


FIG. 4.1 – Les différents niveaux de certification CMMI

4.3.3 Procédure de certification

Les entreprises souhaitant faire reconnaître et communiquer leur niveau de maturité CMMI doivent soumettre leurs processus à une évaluation SCAMPI [36]. Cette approche intègre des examinateurs certifiés du SEI formés à l'évaluation de maturité organisationnelle. Les résultats définitifs indiquent la notation obtenue (niveau de maturité) ainsi que la description des forces et opportunités d'amélioration de chaque domaine de processus du périmètre évalué.

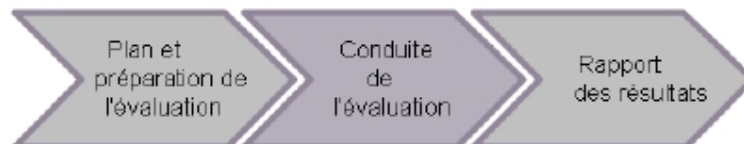


FIG. 4.2 – Phases d'évaluation SCAMPI

Une évaluation SCAMPI repose comme indiqué dans la figure 4.2 sur trois phases décrites brièvement ci-dessous. L'ensemble de l'évaluation dure approximativement trois mois avant d'obtenir les résultats.

Planification et préparation de l'évaluation

Lors de cette étape, l'équipe de formation est formée. Elle est composée de quatre à huit personnes ayant suivi des cours d'introduction à CMMI et d'évaluation SCAMPI. Cette équipe est dirigée par un directeur d'évaluation. Le planning détaillé des activités d'évaluation est élaboré lors de cette étape précisant quelles personnes seront rencontrées pour entretien, les projets qui feront l'objet d'une analyse, etc.

Conduite de l'évaluation

Après une présentation du planning et des objectifs de l'évaluation aux interviewés et dirigeants, l'évaluation est menée par l'examen des projets, leur validation et vérification.

L'équipe d'évaluation conduit les interviews des participants des projets sélectionnés puis enregistre ses observations pour en tirer des conclusions au regard des pratiques CMMI.

En fonction du jeu résultant d'observations, l'équipe d'évaluation note les objectifs, domaines de processus et le niveau de maturité des processus de l'entreprise pour finalement rassembler ses conclusions dans une note finale d'évaluation donnant lieu à une présentation.

Le rapport des résultats

Le Directeur d'évaluation présente les conclusions finales à la direction générale, interviewés et autres intervenants concernés le dernier jour de l'intervention sur site. L'entreprise reçoit alors sa notation de maturité ainsi que le détail de ses forces et domaines d'amélioration.

Le Directeur d'évaluation TeraQuest adresse également un rapport final à l'entreprise qui synthétise les résultats afin qu'aucun projet ni individu ne puissent être identifiés comme source des données. Ce rapport confidentiel est finalement archivé auprès du SEI.

4.3.4 Critique

La procédure de certification CMMI des entreprises n'a rien avoir avec celle d'IFPUG certifiant des personnes. Elle est en effet nettement plus longue et onéreuse. Il ne s'agit pas de faire passer au personnel un examen mais bien d'examiner leur façon de travailler et les projets réalisés pour accorder un niveau de certification. Ce genre de procédure pourrait être parfaitement calquée par la certification de la mesure COSMIC-FFP si celle-ci certifiait les entreprises.

Chapitre 5

Conception du processus de certification pour COSMIC

5.1 Introduction

La méthode de mesure fonctionnelle COSMIC-FFP jouit à l'heure actuelle d'une position de choix dans le secteur de la mesure logicielle de par son statut de norme ISO la reconnaissant internationalement, son hétérogénéité dans les types d'application qu'elle mesure, etc. Elle est incontestablement l'une des plus matures des méthodes de mesure fonctionnelle actuelles.

Il n'en va hélas pas de même de la mesure d'un projet bien précis. Rien n'atteste en effet de la compétence du mesureur et donc de la qualité de sa mesure. Or, la méthode COSMIC-FFP se prévalant d'être d'une grande qualité, la certification des mesureurs se situe dans la continuité de sa recherche de qualité. Ce gage de qualité devrait également encourager l'utilisation de cette méthode de mesure par les entreprises alors certaines de disposer de personnes compétentes et ainsi étendre son utilisation.

Les entreprises du secteur IT ont bien compris que leur avenir est intimement lié à la qualité de leurs produits et services. C'est dans ce contexte qu'elles offrent à leurs employés des formations tout au long de leur carrière. Ceux-ci sont d'ailleurs généralement enthousiastes pour les formations et certifications puisqu'ils en sont généralement récompensés. Une étude menée par la société Prometric sur les raisons qu'ont les personnes de se certifier et ce qu'elles en tirent comme avantage principal donnent les résultats illustré aux figures 5.1.

Il sera tout d'abord question dans ce chapitre de la mise en place du processus de certifica-

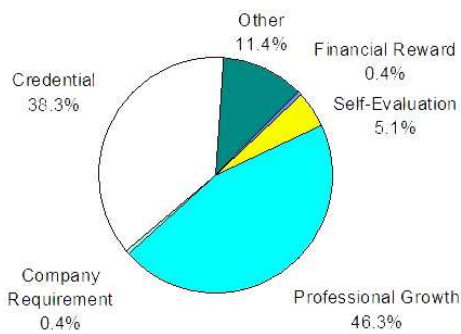


FIG. 5.1 – Principales raisons de se certifier[39]

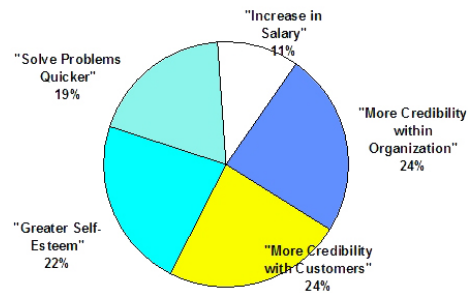


FIG. 5.2 – Principaux avantages retirés d'une certification[39]

tion. Il a non seulement fallu élaborer le matériel pour l'examen en lui-même mais également mettre à jour une stratégie d'organisation de tout le processus de certification. Ensuite, une section sera consacrée au développement du logiciel servant à l'administration des candidats, leurs résultats, etc. On disposera alors des informations nécessaires pour pouvoir procéder à une comparaison du processus de certification avec les entités extérieures analysées dans le chapitre précédent afin de relever les forces et faiblesses et de voir ce qu'il reste à accomplir.

5.2 Organisation du processus de certification

La mise en place du processus de certification est loin d'être une chose rapide et aisée. Les premiers travaux ont officiellement débuté lors d'une rencontre du groupe COSMIC à Berlin le 3 novembre 2004. Un groupe d'étude fut établi pour préparer des recommandations au groupe COSMIC s'interrogeant sur les besoins d'une certification pour les mesureurs utilisant la norme ISO 19761. Ce groupe était composé des Professeurs A. Abran (Canada, ETS), N. Habra (Belgique, FUNDP) et J. Dolado (Espagne, UPV). Leur rapport ayant pour titre « International Certification of measurers using ISO 19761 : 2003 COSMIC Measurement Method »[18] fut remis le 6 juin 2005. Il y est question d'un relevé des points généraux devant être traités pour mettre en place le système de certification. Les points qui vont suivre font état de l'avancement actuel du processus suite à mon stage.

5.2.1 Deux niveaux de certification

L'épreuve de certification de COSMIC-FFP est organisée en deux niveaux afin de pouvoir satisfaire le plus de personnes et leur offrir un examen adapté.

Niveau de base (Entry level)

En théorie, ce niveau de certification demande une connaissance des règles de mesure COSMIC-FFP. Les candidats seront évalués sur le manuel de mesure COSMIC[21] et la norme ISO 19761[3] excepté la partie de la matière relative aux couches logicielles. Les mises à jour de la norme (Update Bulletins) font également partie de la matière.

En pratique, les mesures demandées se feront sur base d'exigences fonctionnelles bien détaillées et spécifiées. Ce seront des cas d'étude bien connus et assez simples.

Niveau expert (expert level)

Pour la théorie, les candidats experts devront maîtriser les règles de mesures COSMIC-FFP. En plus de la norme ISO 19761 et du manuel de mesure complet, des questions seront posées sur une série de publications relatives à la méthode de mesure COSMIC-FFP.

En pratique, des mesures devront être calculées pour des exigences fonctionnelles de diverses qualités.

5.2.2 Public cible

Certification de base

La certification des mesureurs COSMIC-FFP est essentiellement destinée aux praticiens et aux étudiants en informatique ou d'une discipline associée.

Les praticiens sont les personnes qui exercent déjà des mesures sur des logiciels dans leur métier par exemple. Ils pourront ainsi bénéficier d'une reconnaissance attestant de leur compétence en tant que mesureur COSMIC-FFP et donc implicitement de la qualité de leurs mesures. Les étudiants d'un cursus universitaire peuvent également se voir proposés dans le cadre de leur formation cette certification dans le cadre d'un cours de mesure.

Certification au niveau expert

Les praticiens expérimentés dans la méthode de mesure fonctionnelle COSMIC-FFP.

5.2.3 Pré-requis pour l'admission à l'examen

Certification de base

Aucun pré-requis n'est nécessaire pour la certification de base si ce n'est qu'il est préférable d'avoir déjà réalisé quelques exercices de mesure pour une bonne préparation à l'épreuve.

Certification au niveau expert

L'introduction d'un dossier sera exigée afin de pouvoir être admis à l'épreuve de certification. Ce dossier devra contenir la preuve d'avoir une expérience dans la mesure d'au moins 2000 Cfsu. Les applications mesurées devront être de différents types¹ et de plusieurs couches.

5.2.4 Le matériel d'examination

A l'heure actuelle, seul le matériel pour la certification de base a pu être créé. Il se compose d'une part d'une banque de plus de deux cent questions en anglais avec les réponses associées et d'autre part, de cas bien maîtrisés qui serviront de matériel de mesure.

Avant chaque examen, les questionnaires auront été réalisés avec des questions générées aléatoirement. Pour ce faire, la matière a été divisées en chapitre dans lesquels une série de questions ont été rédigées. Toutes ces questions ont été entrées dans un logiciel libre appelé « Questions-Réponses » qui génère des questionnaires en prenant un certain nombre de questions au hasard dans chaque chapitre.

Nous pouvons dès lors être certains que l'épreuve théorique portera sur toute la matière et que les examens seront différents d'une fois à l'autre.

Un questionnaire théorique est constitué de trois parties :

- des questions ouvertes :

Question : « A functional process sends a data group on the Internet. But the network is congested and the network layer using TCP needs to send the data group four times before finally to receive an acknowledgement character. Enumerate all data movement(s). »

Réponse : *Only one : an Exit. An Exit includes all data formatting and presentation manipulations required by the users, including processing required for routing the output to these users.*

- des questions Vrai-Faux :

Question : « Each data group must be directly related to 1 object of interest described in the software's functional user requirements »

Réponse : *True*

- des questions à choix multiples :

Question : « An entry always receives data attributes ... »

... from a single data group. (*True*)

... from at least one data group. (*False*)

¹logiciel de gestion, logiciel en temps-réel ou hybride

... from a single data group inside the boundary. (*False*)

5.2.5 Calibrage des questionnaires

Une fois les questions élaborées, un examen fictif de certification a pu être organisé le 13 décembre 2005 à l'ETS. Une journée de formation fut proposée aux six candidats inscrits à l'épreuve. Ces candidats sont des étudiants diplômés (master degree) en informatique. Seuls trois d'entre-eux ont passé effectivement l'examen. Les modalités étaient les suivantes :

- partie théorique : 29 questions ouvertes — 46 questions Vrai ou Faux — 9 questions choix multiples
- durée de l'épreuve : deux heures
- Ils avaient droit au manuel de mesure COSMIC-FFP pendant l'épreuve.
- pas de partie pratique

Les résultats obtenus après corrections étaient de 52%, 71% et 75%.

Pour ce qui a trait aux questions, il est apparu que certaines questions étaient ambiguës puisque les réponses apportées n'étaient pas assez précises. Elles ont fait l'objet d'une révision. Suite à ces résultats, le seuil de réussite envisagé pour réussir la certification de base s'élève à 75% et celle des experts à 90%.

5.2.6 Validité de la certification

La certification est valable deux ans après quoi il faut repasser l'examen. De cette façon, le niveau des mesureurs reste constant et permet d'accroître la confiance en cette certification.

5.2.7 Organisation

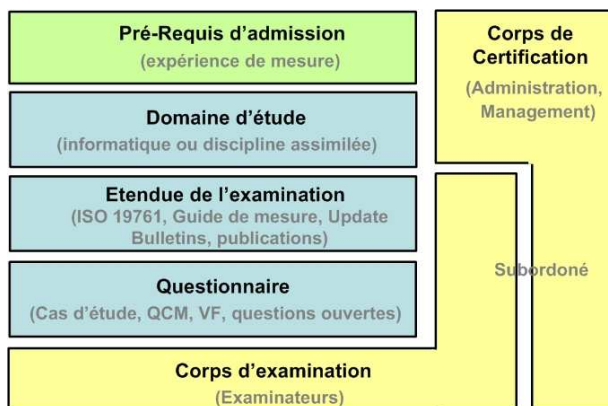


FIG. 5.3 – Structure de l'examen

La figure 5.3 illustre le rôle des deux entités qui seront chargées de la certification de COSMIC-FFP.

Les examinateurs gèrent tout ce qui a trait à l'examen. Ils seront supervisés par le corps de certification qui aura également à sa charge la réalisation des questionnaires, les admissions aux examens et l'administration du dossier des candidats.

5.3 Logiciel d'administration du processus de la certification

Un logiciel a dû être développé afin de pouvoir satisfaire les besoins d'administration de la procédure de certification. Cette section est consacrée à l'analyse de ce dernier. Des captures d'écran de l'application ont été réalisées et font l'objet de l'annexe C.

5.3.1 Vue d'ensemble

Le système d'information se divise en deux parties distinctes interagissantes. La première application gère tout ce qui touche de près à la gestion des candidats, à leurs inscriptions et aux épreuves d'examen.

La seconde partie vise à concevoir un site web qui publierait les noms des personnes certifiées. Ce site web pourrait également renseigner les *examineurs* sur les personnes inscrites aux futures épreuves et récupérer leurs coordonnées.

Seule la première partie, prioritaire, a été développée à ce jour. Elle inclut donc une catégorie d'utilisateurs « *examineur* » en prévision de la conception du site web leur servant de portail à diverses opérations. Ces deux applications partageront la même base de données, ce sera leur seul lien.

5.3.2 Profil des utilisateurs

L'administrateur

Le seul utilisateur habilité à utiliser le système est l'administrateur. Celui-ci sera investi par le comité COSMIC de toute l'administration des candidats, examinateurs et administrateurs du processus de certification. Il peut réaliser toutes les opérations représentées dans le diagramme

des cas d'utilisation repris ci-dessous et décrites dans les scénarii se trouvant en annexe A.

L'administrateur n'a pas besoin de pré-requis pour pouvoir utiliser le système. Etant donné son rôle critique et central dans l'organisation de la certification, cette personne devra être une personne intègre remplissant correctement les tâches qui lui sont dévolues.

L'examineur

Les examinateurs ont été pris en compte dans un soucis d'évolutivité du logiciel. Il est en effet plausible de devoir à l'avenir leur accorder certains droits d'accès à l'information telle que par exemple les résultats précis obtenus par les candidats ou simplement leurs coordonnées.

Le candidat

Les candidats sont les personnes pour lesquelles le système est mis en place. Ce sont leurs informations que le logiciel traitera. Ils pourront avoir accès via un site web aux informations concernant leur acceptation à une épreuve de certification et plus tard aux résultats (réussite ou non) de ces épreuves.

5.3.3 Diagramme des Use Cases

Le diagramme des Use Cases est illustré à la figure 5.4. Les scénarii liés à chacun de ces cas d'utilisation se trouvent en annexe A.

5.3.4 Exigences non fonctionnelles

Evolutivité — Maintenance

- L'application est développée sur base d'une architecture (cfr section ci-dessous) et d'une documentation complète du système facilitant la mise à jour et l'évolution du système.
- Le système est prévu pour pouvoir fonctionner ultérieurement avec d'autres acteurs tels que les examinateurs.
- Le système est actuellement prévu pour pouvoir fonctionner avec une seule console. Cependant, si l'utilité de plusieurs consoles d'administrateur venait à se justifier, le système reste extensible en ajoutant l'utilisation du protocole d'accès concurrentiel à la BD « *two phase commit* » et de légères modifications du système.

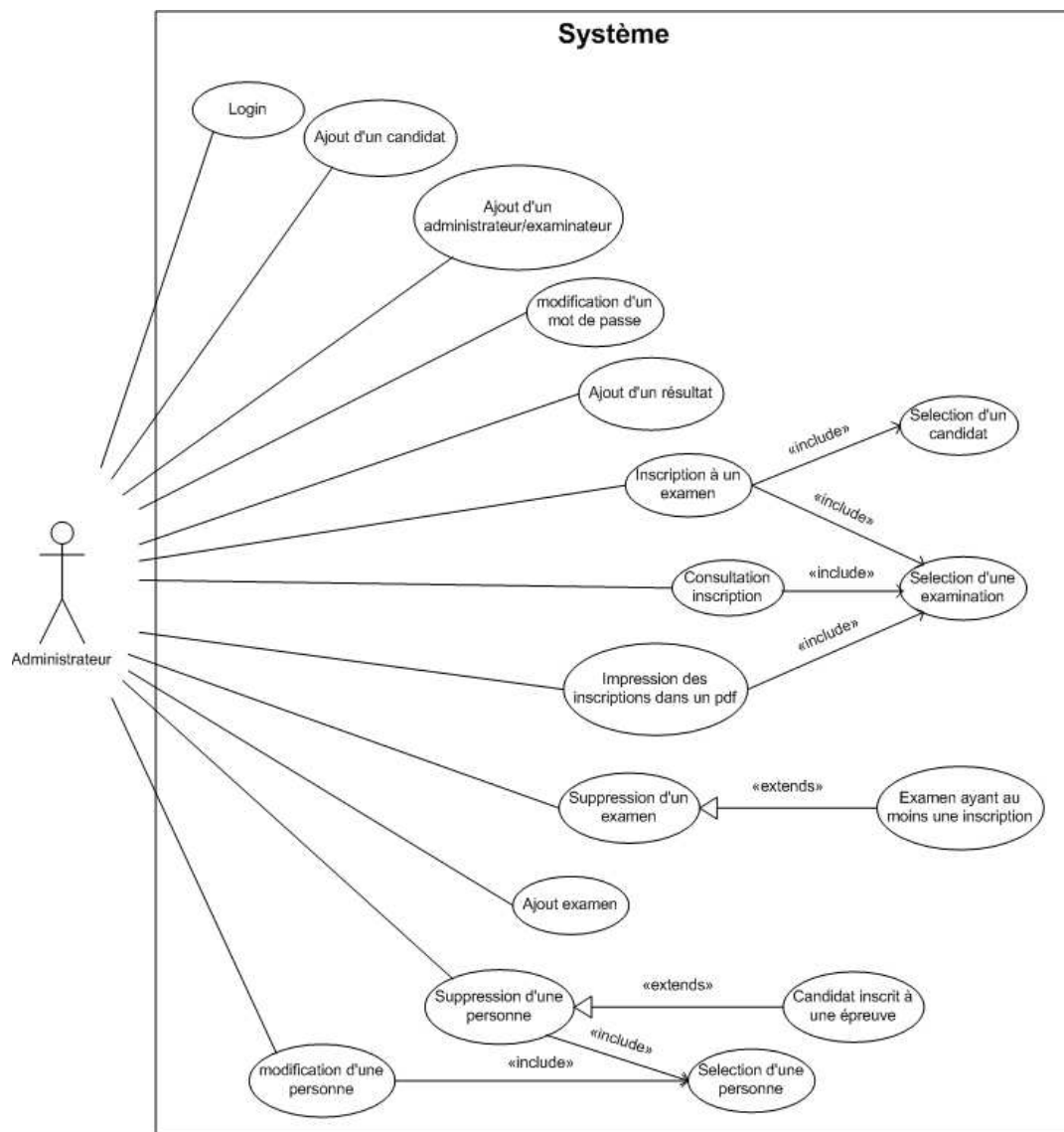


FIG. 5.4 – Use Cases Diagram

Performant

- Le système a des temps de réponse quasi instantanés. Les performances sont optimisées notamment par une connexion unique et permanente à la base de données.

Fiabilité des données

- Le système vérifie la complétude des informations entrées et leur cohérence lorsque qu'il en a la possibilité. Par exemple, une date d'une nouvelle épreuve de certification ne pourra être antérieure à la date actuelle.

Sécurisé

- Toutes les informations persistantes sont stockées dans une base de données de type SQL Server distante et sécurisée. Son accès via l'application demande un login et mot de passe d'administrateur.

Interfaçable avec d'autres systèmes

- Lorsqu'un site web sera construit afin d'offrir un accès aux informations, il pourra avoir accès aux informations stockées dans la base donnée du système existant.

5.3.5 Pattern d'architecture

L'architecture du logiciel adopte le pattern dit « *Modèle-Vue-Contrôleur* ». Le méta-modèle de ce type d'architecture est illustré à la figure 5.5.

Le *Modèle* représente le comportement de l'application : traitements des données, interactions avec la base de données, etc. Il décrit les données manipulées par l'application. Il s'agit donc du coeur de l'application représenté par le diagramme de classe dans la section suivante.

La *Vue* correspond à l'interface avec laquelle l'utilisateur interagit. Les résultats renvoyés par le modèle sont dénués de toute présentation gérée uniquement par les vues. La vue n'effectue aucun traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle, et de permettre à l'utilisateur d'interagir avec elles. Il s'agit donc dans ce cas de toutes les classes d'interface.

le *Contrôleur* prend en charge la gestion des événements pour mettre à jour la vue ou le modèle. Il n'effectue aucun traitement, ne modifie aucune donnée. Il analyse la requête du client

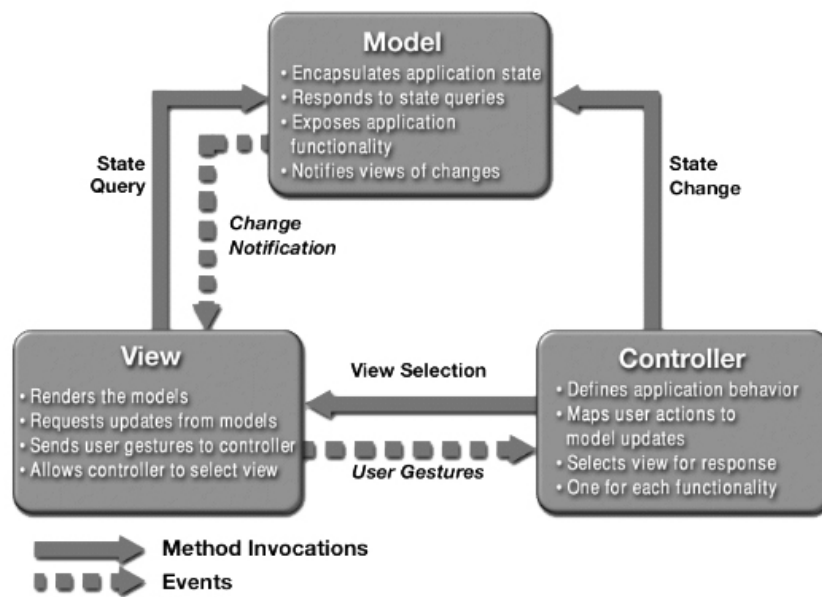


FIG. 5.5 – MVC pattern[33]

et se contente d'appeler le modèle adéquat puis de renvoyer la vue correspondant à la demande.

Du point de vue des fonctionnalités, ces éléments forment trois niveaux différents qui s'enchaînent. Les interconnexions entre les couches se limitent au strict minimum pour éviter les dépendances. Ce dernier point assure qu'il est possible de modifier une couche sans devoir se préoccuper des répercussions sur les deux autres.

5.3.6 Diagramme de classe

La figure 5.6 représente le diagramme de classe de l'application. Le paradigme de programmation choisi pour cette application est l'Orienté-Objet implémenté via la technologie Java.

La partie contrôleur de l'application est en relation avec le modèle par trois classes servant de lien. Ces trois classes sont conçues avec un pattern de conception *Singleton*. Celui-ci garanti l'unicité d'instanciation de ces classes. Il s'agit des classes de gestion nommées *GestionExamen*, *GestionInscription* et *GestionPersonne*. Elles manipulent leurs objets respectifs² répertoriés dans un vecteur par l'intermédiaire de méthodes *getter* et *setter*³, les attributs étant privés. La classe *SQL* est également conçue avec ce pattern. La connexion avec la base de données sera

²par exemple, la classe *GestionInscription* répertorie toutes les instances de la classe *Inscription* à l'aide du vecteur *listInscription*

³non représentées dans le diagramme de classe pour une question de lisibilité

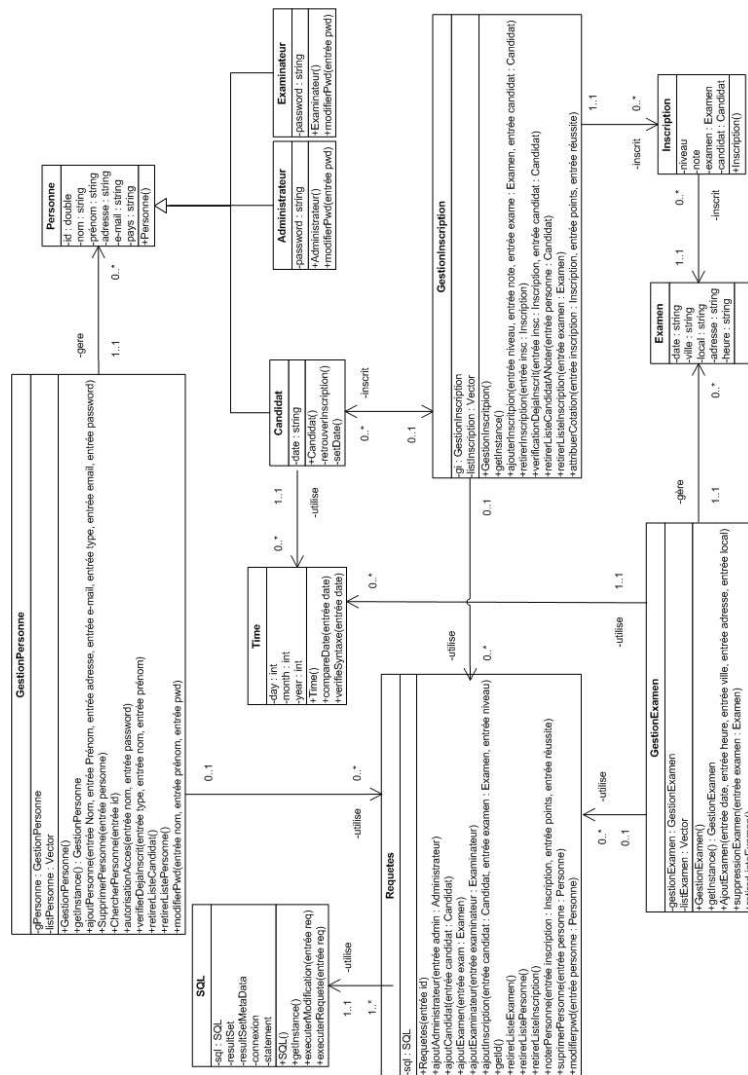


FIG. 5.6 – Diagramme de classe

ainsi effectuée lors de la première requête (i.e. la vérification du mot de passe de l'administrateur) puis restera « ouverte ». Les requêtes ultérieures utiliseront la même connexion offrant ainsi un gain de performance.

Dans un souci d'évolutivité de la base de données, toutes les requêtes SQL se situent dans une seule et même classe (classe Requetes). S'il fallait à l'avenir passer à une base de données oracle ou autre, seule cette classe devrait être modifiée ainsi que celle de connexion (classe SQL).

5.3.7 Diagrammes de séquence

Les trois diagrammes de séquence suivants illustre la manière dont est implémentée l'application.



Nouvelle épreuve de certification

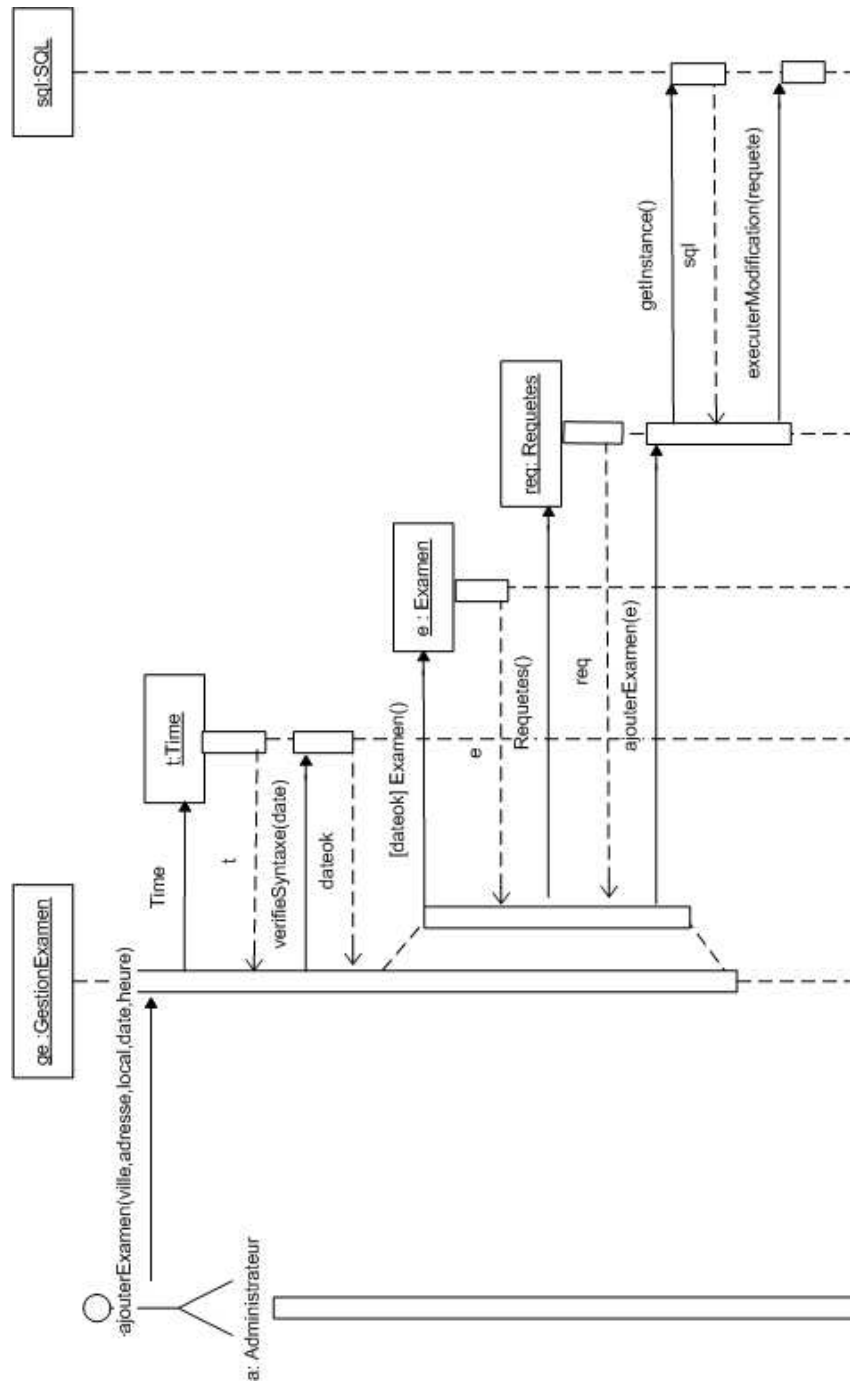


FIG. 5.8 – Diagramme de séquence d'ajout d'un nouvel examen

Inscription d'un candidat à une épreuve

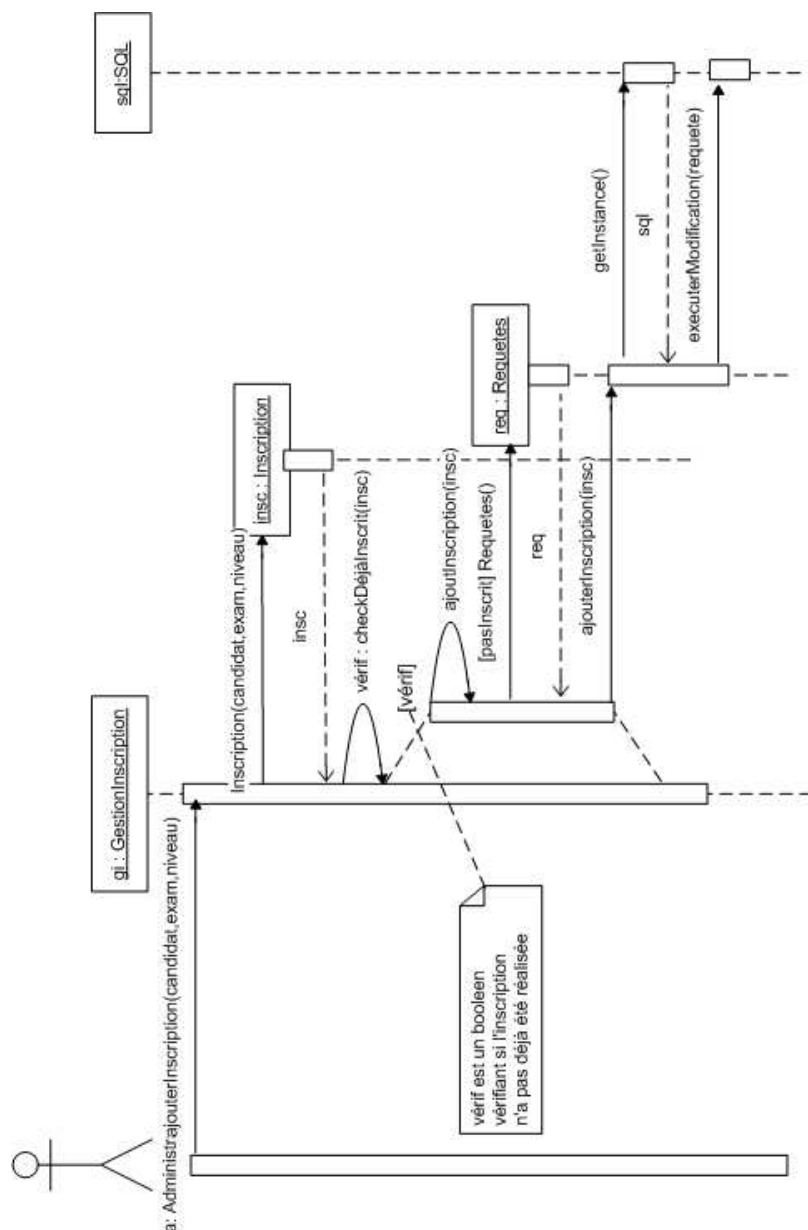


FIG. 5.9 – Diagramme de séquence d'ajout d'une inscription

5.4 Comparaison avec l'existant

Le processus de certification COSMIC-FFP mis en place à l'heure actuelle tend à correspondre au mieux aux types d'organisation pratiquant déjà une certification. L'expérience de celles-ci est un atout, l'analyse de leur mode opératoire permet de retirer les points forts et les faiblesses de chacune afin de construire une nouvelle organisation de certification adaptée à

COSMIC-FFP.

C'est dans cette optique que le groupe COSMIC oeuvre dans le développement du processus de certification. Les différents règlements, formulaires d'inscription, formulaires de demande d'hébergement d'une session d'examen, etc sont déjà pratiquement pré-formulés.

Il en va de même pour ce qui est de la structure de l'examen illustrée par la figure 5.3 qui s'inspire du célèbre centre d'examen d'ingénieurs en information et technologie du Japon (IPA).

En outre, les procédures d'examen adoptées par IFPUG correspondent parfaitement à celles envisageables pour COSMIC. C'est pourquoi de fortes ressemblances peuvent être observées vis-à-vis des pré-requis demandés, de la période de validité ou encore des questionnaires.

5.5 Conclusion

Bien que n'ayant pas encore fait ses preuves, le processus de certification COSMIC-FFP semble être au point pour la partie théorique de la certification de base. Il faudra cependant consolider la banque de questions existantes afin d'en disposer en quantité suffisante. Il ne reste alors plus qu'à sélectionner les personnes qui auront un rôle à jouer au sein de l'organisation.

Des examinateurs vont devoir être formés ainsi que des personnes capables de donner des séances préparatoires à l'examen ; un examinateur ne pouvant bien évidemment pas dispenser ce cours et vice versa. Des exercices de mesure devront aussi être construits pour la partie pratique du niveau expert, ceux-ci devant être d'un niveau de difficulté plus élevé. Les spécifications seront moins précises tout en restant parfaitement maîtrisées par les correcteurs. Par ailleurs, un certain nombre de questions restent encore en suspens telles que les tarifs qui seront demandés, la date de la première épreuve officielle de certification, etc. Ceci demande avant tout que le projet en lui-même soit entériné par le groupe COSMIC.

La prochaine étape devrait se dérouler en mai 2006 à Rome où un pré-test officiel sera planifié durant la conférence *Software Measurement European Forum 2006* (SMEF2006).

Quatrième partie

4^ePartie : Conclusion et perspectives à venir

Chapitre 6

Conclusion

La rapidité et la qualité sont deux critères prépondérants dans tous les secteurs de notre société. Les slogans convergent tous en ce sens : « *High performance. Delivered* » (Accenture), « *Quality in Everything We Do* » (Ernst & Young), « *Quality & Speed* » (foxconn), etc. Le secteur de l'informatique ne déroge pas à la règle. On peut conclure que ce mémoire s'est axé sur ces deux principes décisifs afin de conforter la méthode de mesure fonctionnelle COSMIC-FFP.

L'objectif de rapidité a été atteint par la recherche d'un modèle réduisant considérablement la phase de mesure fonctionnelle procurant ainsi un gain de temps incontestable. Malgré une relative imprécision, le modèle de la régression linéaire multiple confère une très bonne estimation. Ce genre de recherche devrait permettre à la phase de mesure de mieux s'intégrer dans le cycle de développement du logiciel. De cette manière, les prévisions du budget et de l'échéancier du projet ne peuvent être que plus précises augmentant ainsi la confiance des clients et leur fidélité.

La démarche de recherche du meilleur modèle entreprise est basée sur une analyse d'un échantillon de mesure parfaitement maîtrisé. Un large panel d'algorithmes de classification a permis de faire ressortir le meilleur modèle. La comparaison et l'attestation de qualité ont pu être réalisées sur base de taux d'erreur.

L'objectif de qualité a pour sa part été rempli suite à la mise en place du processus de certification. Celui-ci garanti la compréhension et la maîtrise de la méthodologie de la mesure COSMIC-FFP par les personnes certifiées diminuant ainsi les risques inhérents à une application mal mesurée. Les organisations de certification existantes ont contribué à la mise en place d'un système de certification efficace et opérationnel pour le niveau d'examen de base.

Chapitre 7

Perspectives à venir

Les résultats fournis par la technique de régression linéaire et le processus de certification ne sont pas encore arrivés à maturité. Pour sa part, l'utilisation du modèle de régression estimant la mesure fonctionnelle selon la méthode COSMIC-FFP n'a pas encore fait ses preuves. Une série d'autres projets bien mesurés devraient être testés afin d'attester ou non de sa précision et qualité. Au besoin, il pourrait être utile de mettre le modèle à jour sur base de ces nouvelles mesures et plus spécialement sur base des use cases de grande taille qui représentent l'actuel point faible de la régression. Par ailleurs, bien qu'un nombre important d'algorithmes aient été testés, l'existence d'un meilleur modèle est toujours possible.

Le processus de certification n'est lui aussi qu'à ses débuts. En effet, seul le niveau de certification de base est réalisé à l'heure actuelle. Le niveau d'expert requiert l'ébauche de nouvelles questions théoriques et pratiques plus difficilement réalisables. En outre, une campagne publicitaire devraient être menée au sein des universités et entreprises afin de faire connaître cette nouvelle certification.

Le logiciel d'administration doit également faire l'objet de l'ajout d'un site internet procurant certains services aux candidats et examinateurs.

Bibliographie

- [1] Norme ISO/IEC 14143-3 :2003. Information technology — software measurement — functional size measurement — part 3 :verification of functional size measurement methods. *Geneva : International Organization for Standardization*, 2003.
- [2] Norme ISO/IEC 14143 :1997. Information technology — software measurement — functional size measurement — definition of concepts. 1997.
- [3] Norme ISO/IEC 19761 :2003. Software engineering — cosmic-ffp — a functional size measurement method. *Geneva : International Organization for Standardization*, 2003.
- [4] Khelifi A., Abran A., Simons C., and Desharnais J-M. Hotel accommodation system (reservation). *Software Engineering Research Laboratory*, February 2005.
- [5] Khelifi A., Abran A., Simons C., and Desharnais J-M. Valve control system. *Software Engineering Research Laboratory*, February 2005.
- [6] Khelifi A., Abran A., Simons C., and Desharnais J-M. Automatic line switching. *Software Engineering Research Laboratory*, March 2005.
- [7] Khelifi A., Abran A., Simons C., and Desharnais J-M. L-euchre card game system (minimal implementation). *Software Engineering Research Laboratory*, March 2005.
- [8] Khelifi A., Abran A., Simons C., and Desharnais J-M. Rice cooker system. *Software Engineering Research Laboratory*, March 2005.
- [9] Khelifi A., Abran A., Simons C., and Desharnais J-M. C-registration system. *Software Engineering Research Laboratory*, May 2005.
- [10] Khelifi A., Abran A., Simons C., and Desharnais J-M. Collegiate sports paging system. *Software Engineering Research Laboratory*, May 2005.
- [11] Khelifi A., Abran A., Simons C., and Desharnais J-M. Savat gateway (saga) system. *Software Engineering Research Laboratory*, May 2005.
- [12] Alain Abran. New views — ffp concepts. NASSCOM QUALITY SUMMIT, Bangalore (India), sept 2004. powerpoint presentation.

- [13] Alain Abran and Jean-Philippe Jacquet. From software metrics to software measurement methods : A process model. *International Software Engineering Standards Symposium and Forum*, pages 128–135, 1997.
- [14] AI Access, 2005. http://www.aiaccess.net/f_accueil.htm, accédé le 05 avril 2006.
- [15] Abran Alain. *Analysis of the measurement process of function point analysis*. PhD thesis, Université de Montréal, Mars 1994.
- [16] Abran Alain. New views : Ffp concepts — cosmic-ffp — iso 19761 the 2nd generation of functional size measurement. *NASSCOM QUALITY SUMMIT*, 2004.
- [17] Tisserant Alain. La qualité du logiciel. <http://www.mines.inpl-nancy.fr/~tisseran/cours/qualite-logiciel/consequences.html>, accédé le 11 décembre 2006, 2004.
- [18] Naji Habra Alain Abran and Javier Dolado. International certification of measurers using iso 19761 : 2003 cosmic measurement method. *Study Group Report*, 28 février 2005.
- [19] Albrecht Allan. Measuring application development productivity. *Proceedings IBM Applications Development Symposium*, pages 34–43, 1979.
- [20] Albrecht Allan. Productivity measurement and estimate validation. *IBM Corporate Information Systems and Administration*, 1985.
- [21] COSMIC CORE TEAM AUTHORS. Measurement manual (the cosmic implementation guide for iso/iec 19761 :2003. 2003.
- [22] Saadi Azzouz and Alain Abran. A proposed measurement role in the rational unified process and its implementation with iso 19761 : Cosmic-ffp, 2004. <http://www.internet.uqam.ca/web/t15040>, accédé le 20 mars 2006.
- [23] IFPUG Certification Committee. Ifpug regional cfps examination rules and responsibilities. 2005.
- [24] Jean Debord. Informatique et chimie, 2001. http://www.unilim.fr/pages_perso/jean.debord/math/reglin/reglin.htm, accédé le 10 mars 2006.
- [25] Certified Fund Raising Executive. Becoming a cfre power point presentation. <http://cfre.org>, accédé le 03 janvier 2006, 2005.
- [26] Le Lann Gérard. The ariane 5 flight 501 failure — a case study in system engineering for computing systems. 1996.
- [27] The International Function Point UsersGroup. <http://www.ifpug.org/>, accédé le 09 décembre 2005, 2005.
- [28] International Function Point Users Group (IFPUG). Function points counting practices manual — release 4.0. *Ohio : International Function Point User Group (IFPUG)*, 1994.

- [29] Capers Jones. The spr feature point method. *Software Productivity Research Inc.*, 1986.
- [30] Lord Kelvin. Popular lectures and addresses. 1889.
- [31] Rachid Ladjadj, 2003. <http://www-igm.univ-mlv.fr/~dr/XPOSE2002/Neurones/index.php?rubrique=Accueil>, accédé le 06 avril 2006.
- [32] Richard Lepage. Gpa779 : Application des systèmes experts et des réseaux de neurones, 2005. www.gpa.etsmtl.ca/cours/gpa779/Documents/Cours8.ppt, accédé le 06 avril 2006.
- [33] Sun Microsystems. Java blueprints — model-view-controller, 2000-2002. <http://java.sun.com/blueprints/patterns/MVC-detailed.html>, accédé le 10 avril 2006.
- [34] H. Mintzberg. *Structure et dynamique des organisations*. Les Editions d'organisation, 1994.
- [35] Monique Noirhomme. Syllabus de statistique et probabilité : 2ème bac, 2003.
- [36] Members of the Assessment Method Integrated Team. Standard cmmism appraisal method for process improvement (scampism), version 1.1 : Method definition document. *CMU/SEI-2001-HB-001*, December 2001.
- [37] St-Pierre D. Maya M. Abran A. Desharnais J.-M. Bourque P. Full function points : Counting practices manual. *Technical Report 1997-04*, 1997.
- [38] Kovarik Patrick. Pannes informatiques : la sncf va renforcer son système de vente de billets. <http://french.epochtimes.com/news/4-12-4/216.html>, accédé le 11 décembre 2005, 2004.
- [39] Drake Prometric. Dataquest study, 2005. www.prometric.com, accédé le 10 avril 2006.
- [40] Meli R., Abran A., Ho V.T., and Oligny S. On the applicability of cosmic-ffp for measuring software throughout its life cycle. *ESCOM-SCOPE 2000*, pages 128–135, April 2000.
- [41] Charles Symons. Function point analysis : Difficulties and improvements. *IEEE Trans Software Eng.*, 14, 1988.
- [42] Inc. The Standish Group International. 2004 third quarter research report. 2004.
- [43] Carnegie Mellon University. <http://www.sei.cmu.edu/>, accédé le 23 décembre 2005, 2005.
- [44] Carnegie Mellon University. Sei credentials program : Certificates vs. certifications. <http://www.sei.cmu.edu/activities/credentials/difference.html>, accédé le 15 décembre 2005, 2005.
- [45] Paul Vickers. An introduction to function point analysis. 2003.
- [46] Scott Whitmire. 3d function points : Scientific and real-time extensions to function points. *Proceedings of the 1992 Pacific Northwest Software Quality Conference*, 1992.

- [47] Wikipedia, 2005. http://fr.wikipedia.org/wiki/R%C3%A9seau_de_neurones, accédé le 05 avril 2006.
- [48] Ian H. Witten and Eibe Frank. Data mining : Practical machine learning tools and techniques. *Morgan Kaufmann*, 2005. <http://www.cs.waikato.ac.nz/~ml/weka/>, accédé le 04 avril 2006.
- [49] T. WONNACOTT and R. WONNACOTT. *Statistique*. ECONOMICA, 1991. 4^e édition.

Annexe A

Scénarii relatifs aux cas d'utilisation

A.1 UC1 Login

Brève description : Ce use case vérifie que la personne désirant utiliser l'application est bien *administrateur* et que son identifiant et mot de passe sont corrects.

Pré : /

Post : Le système est opérationnel.

Administrateur	Système
1. L'administrateur entre son nom et son mot de passe puis clique sur <i>OK</i> .	2. le système vérifie l'identité. 3. Le système autorise l'accès à la fenêtre principale de l'application.

Alternative : Si le nom et/ou le mot de passe introduit sont incorrects, le système le notifie et invite la personne à recommencer.

A.2 UC2 Ajout d'un candidat

Brève description : Ajoute un candidat en lui créant un dossier.

Pré : Le système est opérationnel.

Post : Le candidat est ajouté et enregistré.

Le système est opérationnel.

Administrateur	Système
<p>1. L'administrateur clique sur le bouton d'ajout d'un candidat</p> <p>3. L'administrateur entre les nom, prénom, adresse et e-mail du nouveau candidat puis clique sur <i>OK</i></p>	<p>2. le système affiche une page d'information devant être complétée concernant le nouveau candidat.</p> <p>4. Le système enregistre les informations et ajoute un numéro identifiant.</p>

Alternative : Si un champ devant être rempli est manquant, le système affiche un message d'erreur et invite l'administrateur à compléter.

Si le candidat est déjà inscrit, le système affiche un message d'erreur.

A.3 UC3 Ajout d'un administrateur/examineur

Brève description : Ajoute un administrateur ou examinateur en lui créant un dossier et un accès protégé par mot de passe.

Pré : Le système est opérationnel.

Post : L'administrateur/examineur est ajouté et enregistré.

Le système est opérationnel.

Administrateur	Système
1. L'administrateur clique sur le bouton d'ajout d'un administrateur/examineur	
3. L'administrateur entre les nom, prénom, adresse, e-mail et mot de passe du nouveau candidat puis clique sur <i>OK</i>	2. le système affiche une page d'information devant être complétée concernant le nouvel administrateur/examineur. 4. Le système enregistre les informations et ajoute un numéro identifiant.

Alternative : Si un champ devant être rempli est manquant, le système affiche un message d'erreur et invite l'administrateur à compléter.
Si l'administrateur/examineur est déjà inscrit, le système affiche un message d'erreur.

A.4 UC4 Modification d'un mot de passe

Brève description : L'administrateur modifie le mot de passe d'un administrateur ou examineur.

Pré : Le système est opérationnel.

Post : Le mot de passe d'administrateur ou examineur est modifié et enregistré.
Le système est opérationnel.

Administrateur	Système
<p>1. L'administrateur clique sur le bouton de modification d'un mot de passe</p> <p>3. L'administrateur entre les nom, prénom, nouveau mot de passe et la confirmation du nouveau mot de passe de la personne concernée puis clique sur <i>OK</i></p>	<p>2. le système affiche une page d'information devant être complétée.</p> <p>4. Le système enregistre le nouveau mot de passe de la personne concernée.</p>

Alternative : Si un champ devant être rempli est manquant, le système affiche un message d'erreur et invite l'administrateur à compléter.

Si le mot de passe et sa confirmation sont différents, le système affiche un message d'erreur et invite l'administrateur à recommencer.

A.5 UC5 Sélection d'un candidat

Brève description : Permet de choisir un candidat dans la liste de candidats disponibles.

Pré : le système est opérationnel.

Post : Un candidat est sélectionné.

Le système est opérationnel.

Administrateur	Système
<p>2. L'administrateur sélectionne un candidat puis clique sur <i>OK</i>.</p>	<p>1. le système affiche la liste des candidats enregistrés.</p>

Alternative : Si aucun candidat n'est sélectionné, un message d'erreur demande d'effectuer une sélection.

A.6 UC6 Sélection d'un examen

Brève description : Permet de choisir un examen dans la liste d'examens disponibles.

Pré : Le système est opérationnel.

Post : Un examen est sélectionné.

Le système est opérationnel.

Administrateur	Système
2. L'administrateur sélectionne un examen puis clique sur <i>OK</i> .	1. le système affiche la liste des examens enregistrés.

Alternative : Si aucun examen n'est sélectionné, un message d'erreur demande d'effectuer une sélection.

A.7 UC7 Inscription à un examen

Brève description : L'administrateur inscrit un candidat pour un examen de certification.

Pré : Le système est opérationnel.

Post : Le candidat est inscrit pour l'examen désiré.

Le système est opérationnel.

Administrateur	Système
1. L'administrateur clique sur le bouton d'inscription <i>2. Include UC5 Sélection d'un candidat</i> <i>3. Include UC7 Sélection d'un examen</i> 4. L'administrateur sélectionne le niveau de l'examen désiré puis clique sur <i>OK</i> .	5. Le système enregistre l'inscription.

Alternative : Si le candidat est déjà inscrit à cette épreuve de certification, un message d'erreur informe l'administrateur qu'une inscription a déjà été effectuée pour cet examen.

A.8 UC8 Ajout d'un résultat

Brève description : L'administrateur attribue une note pour un examen à un candidat et indique sa réussite ou non.

Pré : Le système est opérationnel.

Post : Un candidat est évalué pour une inscription à un examen.

Le système est opérationnel.

Administrateur	Système
1. L'administrateur clique sur le bouton d'attribution d'un résultat	
3. L'administrateur sélectionne l'inscription à évaluer, attribue une note et indique la réussite ou non puis clique sur <i>OK</i> .	2. Le système affiche la liste de toutes les inscriptions n'ayant pas encore été notées.
	4. Le système enregistre l'inscription.

Alternative : Si l'administrateur oublie de sélectionner une inscription ou ne remplit pas un champ, le système l'en informe et le prie de recommencer.

A.9 UC9 Consultation des inscriptions

Brève description : L'administrateur désire visualiser les candidats inscrits pour une date et un lieu d'examen.

Pré : Le système est opérationnel.

Post : Une liste de candidats inscrits pour un examen est affichée.

Le système est opérationnel.

Administrateur	Système
1. L'administrateur clique sur le bouton de visualisation des inscriptions.	
2. <i>Include UC7 Sélection d'un examen</i>	2. Le système affiche la liste de tous les candidats inscrits correspondant à l'examen sélectionné.

A.10 UC10 Impression des inscriptions dans un fichier pdf

Brève description : L'administrateur désire générer un fichier pdf contenant les candidats inscrits à un examen.

Pré : Le système est opérationnel.

Post : Un fichier pdf est généré contenant la liste des candidats participant ou ayant participé à un examen.

Le système est opérationnel.

Administrateur	Système
1. L'administrateur clique sur le bouton d'impression d'inscriptions dans un pdf.	
2. <i>Include UC7 Sélection d'un examen</i>	2. Le système génère le fichier contenant la liste des candidats inscrits à l'examen sélectionné.

A.11 UC11 Suppression d'un examen

Brève description : Supprime une examination.

Pré : Le système est opérationnel.

Post : L'examination est supprimé du système. Le système est opérationnel.

Administrateur	Système
1.L'administrateur clique sur le bouton de suppression d'un examen	
2. <i>Include UC7 Sélection d'un examen</i>	
	4.Le système supprime cet examen.

Alternative : Si une candidature est enregistrée pour cet examen, le système affiche un message d'erreur informant l'impossibilité de suppression.

A.12 UC12 Ajout d'un examen

Brève description : Ajoute une date et un lieu pour une épreuve de certification.

Pré : Le système est opérationnel.

Post : L'examen est ajouté au système. Le système est opérationnel.

Administrateur	Système
1.L'administrateur clique sur le bouton d'ajout d'un examen	
3.L'administrateur entre la date et le lieu puis clique sur <i>OK</i>	2.le système demande la date et le lieu.
	4.Le système enregistre ce nouvel examen de certification.

Alternative : Si un champ devant être rempli est manquant, le système affiche un message d'erreur et invite l'administrateur à compléter.

Si le même examen a déjà été enregistré, le système affiche un message d'erreur en informant l'administrateur.

A.13 UC13 Sélection d'une personne

Brève description : Permet de choisir un personne dans la liste de toutes les personnes enregistrées dans le système.

Pré : Le système est opérationnel.

Post : Une personne est sélectionnée.

Le système est opérationnel.

Administrateur	Système
2. L'administrateur sélectionne une personnes puis clique sur <i>OK</i> .	1. le système affiche la liste des personnes.

Alternative : Si aucune personne n'est sélectionnée, un message d'erreur demande d'effectuer une sélection.

A.14 UC14 Suppression d'une personne

Brève description : Supprime une personne du système (candidat, examinateur ou administrateur).

Pré : Le système est opérationnel.

Post : La personne est supprimée du système.

Le système est opérationnel.

Administrateur	Système
1.L'administrateur clique sur le bouton de suppression d'une personne <i>2. Include UC13 Sélection d'une personne</i>	4.Le système supprime la personne sélectionnée.

Alternative : Si la personne est un candidat et qu'il est enregistré pour un examen, le système affiche un message d'erreur informant l'impossibilité de suppression.

A.15 UC15 Modification d'une personne

Brève description : Modifie les informations relatives à une personne (candidat, examinateur ou administrateur).

Pré : Le système est opérationnel.

Post : Les informations de la personne sont mises à jour dans le système.

Le système est opérationnel.

Administrateur	Système
1.L'administrateur clique sur le bouton de modification d'une personne	
2. <i>Include UC13 Sélection d'une personne</i>	
5. L'administrateur modifie les champs désirés puis appuie sur <i>OK</i> .	4.Le système affiche les information de la personne sélectionnée.
	6. Le système met à jour les informations de la personne.

Alternative : Si un champ devant être rempli est manquant, le système affiche un message d'erreur et invite l'administrateur à compléter.

A.16 Dictionnaire des Use Cases

- administrateur : personne ayant l'accès au logiciel d'administration par son nom et son mot de passe. C'est la seule personne abilitée à réaliser les différents UC.
- candidat : personne enregistrée dans le système pouvant s'inscrire à des examens.
- examinateur : personne n'ayant pas encore de rôle au sein du système.
- personne : administrateur ou examinateur ou candidat
- système opérationnel : le système est utilisable par un administrateur correctement connecté. La connexion à la BD est effectuée.
- identité : nom et mot de passe
- dossier : fiche de coordonnées d'une personne (nom, prénom, adresse complète, e-mail et un mot de passe pour un administateur ou examinateur)
- note : pourcentage obtenu lors de l'examen.
- examen : rend compte d'une adresse complète, et de la date de l'épreuve.
- inscription : inscription d'un candidat à un examen.

Annexe B

Comparatif des résultats

Cette annexe a pour but de présenter tous les résultats prédits par les différentes techniques présentées dans le chapitre 3 traitant de la recherche sur l'estimation de la taille fonctionnelle tôt dans le cycle de développement. La première colonne est le nom du Use Case, la deuxième indique la taille COSMIC réelle de ce UC et les suivantes sont les valeurs estimées par les techniques du chapitre 3.

UC	taille réelle (Cfsu)	régresion	réseau de neurones	reptree	reptree élagué	j48	random tree	ridor
Set Target Temperature	5	5,241	5,684	4,349	4,729	5	5	5
Control Heater	4	5,241	5,684	4,349	4,729	5	5	5
Control Indicator Lamp	3	5,241	5,684	4,349	4,729	5	5	5
Restore backup line	5	6,411	6,77	6,346	7,602	5	5	5
Restore Work Line	5	6,411	6,77	6,346	7,602	5	5	5
Remove backup Line	5	6,411	6,77	6,346	7,602	5	5	5
Remove work line	5	6,411	6,77	6,346	7,602	5	5	5
conditional switch to work line	5	6,411	6,77	6,346	7,602	5	5	5
conditional switch to ba- ckup line	5	7,582	7,964	9,023	7,602	12	5	12
forced switch to work line	3	5,241	5,684	4,349	4,729	5	5	5
forced switch to backup line	3	5,241	5,684	4,349	4,729	5	5	5
work line failed	5	5,241	5,684	4,349	4,729	5	5	5
backup line failed	5	5,241	5,684	4,349	4,729	5	5	5
work line degraded	5	5,241	5,684	4,349	4,729	5	5	5
backup line degraded	5	6,411	6,77	6,346	7,602	5	5	5
work line cleared	5	5,241	5,684	4,349	4,729	5	5	5
backup line cleared	5	5,241	5,684	4,349	4,729	5	5	5
subscribe	13	10,373	10,7	12,556	13,375	12	13	12
provide feedback	8	8,032	7,994	4,349	13,375	16	8	4
edit profile	13	10,373	10,7	12,556	13,375	12	13	12
read content on website	14	16,676	18,839	20,333	13,375	14	14	12
pay fee with credit card	12	10,373	10,7	12,556	13,375	12	13	12
post advertising content	17	17,846	20,544	20,333	13,375	14	17	12
print advertiser reports	12	14,334	15,45	20,333	13,375	12	12	12

TAB. B.1: (suite)

UC	taille réelle (Cfsu)	régression	réseau de neurones	reptree	reptree élagué	j48	random tree	ridor
send content	5	5,241	5,684	4,349	4,729	5	5	5
send page	4	5,241	5,684	4,349	4,729	5	5	5
approve story	38	41,165	37,886	20,333		14	38	12
Create reservation	8	8,753	9,263	9,023	7,602	12	8	8
update reservation	12	7,582	7,964	9,023	7,602	12	5	12
accept reservation	8	6,411	6,77	6,346	7,602	5	5	5
confirm reservation	10	7,582	7,964	9,023	7,602	12	5	12
select reservation	8	6,411	6,77	6,346	7,602	5	5	5
cancel reservation	12	7,582	7,964	9,023	7,602	12	5	12
room type report	8	5,241	5,684	4,349	4,729	5	5	5
Hydraulic valve	12	13,435	15,368	9,023	7,602	12	12	8
Host a game	6	5,241	5,684	4,349	4,729	5	5	5
new game	5	6,411	6,77	6,346	7,602	5	5	5
going alone	4	4,07	4,704	4,349	3,667	4	4	4
choosing trump	5	4,07	4,704	4,349	3,667	4	4	4
playing a hand	3	4,07	4,704	4,349	3,667	4	4	4
choose a seat	4	2,899	3,827	4,349	3,667	4	4	4
dealing of cards	3	2,899	3,827	4,349	3,667	4	4	4
joining a game	5	6,411	6,77	6,346	7,602	5	5	5
error message	3	4,07	4,704	4,349	3,667	4	4	4
ending a hand	3	4,07	4,704	4,349	3,667	4	4	4
scoring a hand	5	7,582	7,964	9,023	7,602	12	5	12
winning a trick	4	5,241	5,684	4,349	4,729	5	5	5
winning/loosing a game	5	5,241	5,684	4,349	4,729	5	5	5
lost IP connection	4	2,899	3,827	4,349	3,667	4	4	4
keeping a score	3	4,07	4,704	4,349	3,667	4	4	4
view report card	5	5,241	5,684	4,349	4,729	5	5	5
register for courses	30	23,428	26,997	20,333		12	30	12
select courses to teach	8	6,411	6,77	6,346	7,602	5	5	5
submit grades	6	5,241	5,684	4,349	4,729	5	5	5
maintain professor information	16	15,954	17,176	20,333	13,375	16	16	4
maintain student information	16	15,954	17,176	20,333	13,375	16	16	4
close registration	12	11,544	12,196	12,556	13,375	12	12	12
logon	3	5,241	5,684	4,349	4,729	5	5	5
get account balance	4	4,07	4,704	4,349	3,667	4	4	4
delete product category	12	8,753	9,263	9,023	7,602	12	8	8
get product categ serial nr	4	4,07	4,704	4,349	3,667	4	4	4
logon	7	6,411	6,77	6,346	7,602	5	5	5
product categ list	4	4,07	4,704	4,349	3,667	4	4	4
change pwd	11	7,582	7,964	9,023	7,602	12	5	12
order account statement	7	7,582	7,964	9,023	7,602	12	5	12
register product category	8	7,582	7,964	9,023	7,602	12	5	12
logoff	6	6,411	6,77	6,346	7,602	5	5	5
order goods	7	7,582	7,964	9,023	7,602	12	5	12
freeze logon id	4	5,241	5,684	4,349	4,729	5	5	5
users product category list	5	5,241	5,684	4,349	4,729	5	5	5
adhoc catalog order	3	4,07	4,704	4,349	3,667	4	4	4

TAB. B.1: (suite)

UC	taille réelle (Cfsu)	régression	réseau de neurones	reptree	reptree élagué	j48	random tree	ridor
display statement summary	9	6,411	6,77	6,346	7,602	5	5	5
get account serial nr	4	4,07	4,704	4,349	3,667	4	4	4
list orders for account	10	6,411	6,77	6,346	7,602	5	5	5
get last nr transaction	4	4,07	4,704	4,349	3,667	4	4	4

TAB. B.1: Valeurs estimées par les différentes techniques d'estimation

Le tableau B.2 est un récapitulatif des huit applications ayant servi d'échantillon pour l'analyse avec pour chacune leur taille fonctionnelle (2^e colonne) et la taille approximée par les différentes techniques.

Application	Taille réelle (Cfsu)	régresion	réseau de neurones	reptree	reptree élagué	j48	random tree	ridor
Rice Cooker	12,0	15,7	17,1	13,0	14,2	15,0	15,0	15,0
Automatic Line Switching	66,0	82,7	88,4	77,5	86,3	77,0	70,0	77,0
Collegiate Sports Paging	136,0	139,7	144,2	132,0	103,1	116,0	138,0	98,0
Hotel Reservation	66,0	49,6	52,4	53,1	50,3	63,0	38,0	59,0
Valve Control	12,0	13,4	15,4	9,0	7,6	12,0	12,0	8,0
L-euchre Card Game	62,0	69,2	78,3	73,9	70,0	73,0	66,0	73,0
C-Registration	123,0	112,3	122,8	117,0	84,5	101,0	115,0	73,0
SAVAT-Gateway	82,0	76,3	82,4	81,2	77,3	89,0	61,0	89,0
TOTAL	559,0	559,0	600,8	556,9	493,3	546,0	515,0	492,0

TAB. B.2 – Récapitulatif des estimations par application

Annexe C

Captures d'écran du logiciel

Les captures d'écran ci-dessous illustrent le logiciel d'administration attaché à la certification de COSMIC-FFP.

C.1 Identification de l'administrateur

La capture d'écran C.1 permet à un administrateur d'avoir accès aux fonctionnalités du logiciel. L'interface principale reste inaccessible tant qu'un administrateur valide n'est pas entré.

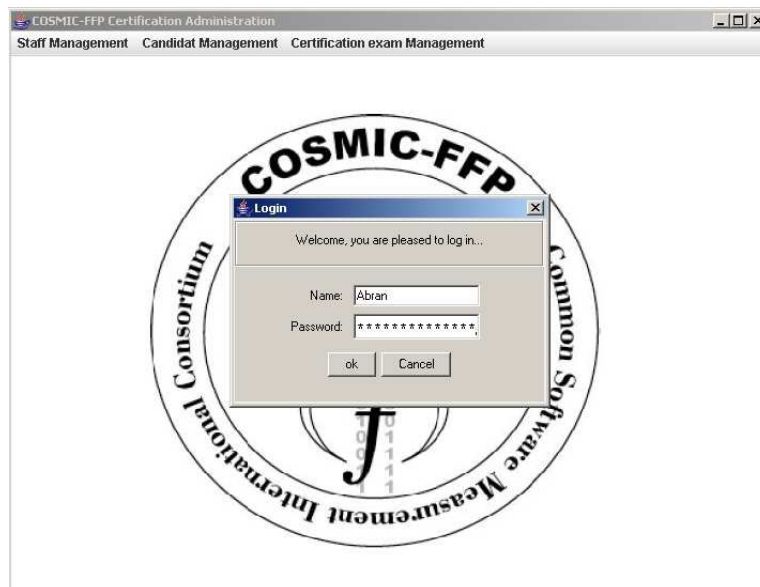


FIG. C.1 – Interface d'identification

C.2 Interface principale

La capture d'écran C.2 représente l'interface principale donnant l'accès à toutes les fonctionnalités du logiciel.

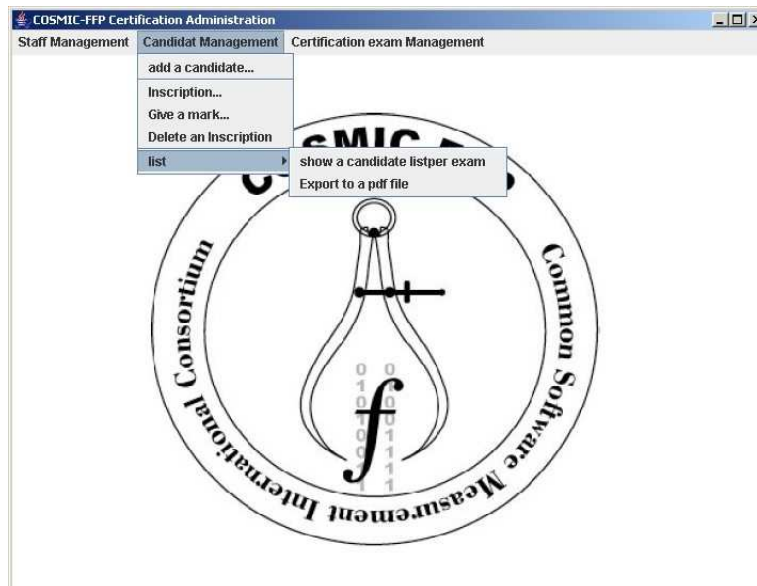


FIG. C.2 – Interface principale

C.3 Interface d'inscription

La capture d'écran C.3 liste tous les candidats et toutes les épreuves d'examens susceptibles de recevoir des inscriptions¹. L'administrateur inscrit un candidat pour une épreuve avec le niveau de difficulté. Si l'une des informations est manquante ou incorrecte², un message d'erreur informe l'administrateur.

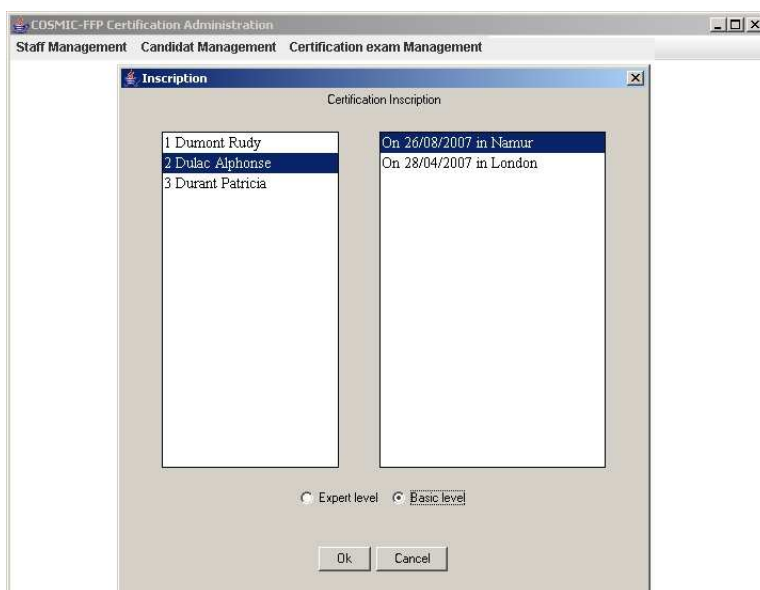


FIG. C.3 – Interface d'inscription à un examen

¹seuls les examens ayant une date postérieure à la date actuelle sont affichés.

²Si le candidat est déjà inscrit à l'épreuve ou s'il est inscrit à la même date mais à l'autre niveau de difficulté.

C.4 Interface d'attribution d'une note

La capture d'écran C.4 liste toutes les inscriptions n'ayant pas encore reçu de cotation. L'administrateur est invité à sélectionner une inscription, lui attribuer une note puis indiquer la réussite ou non.

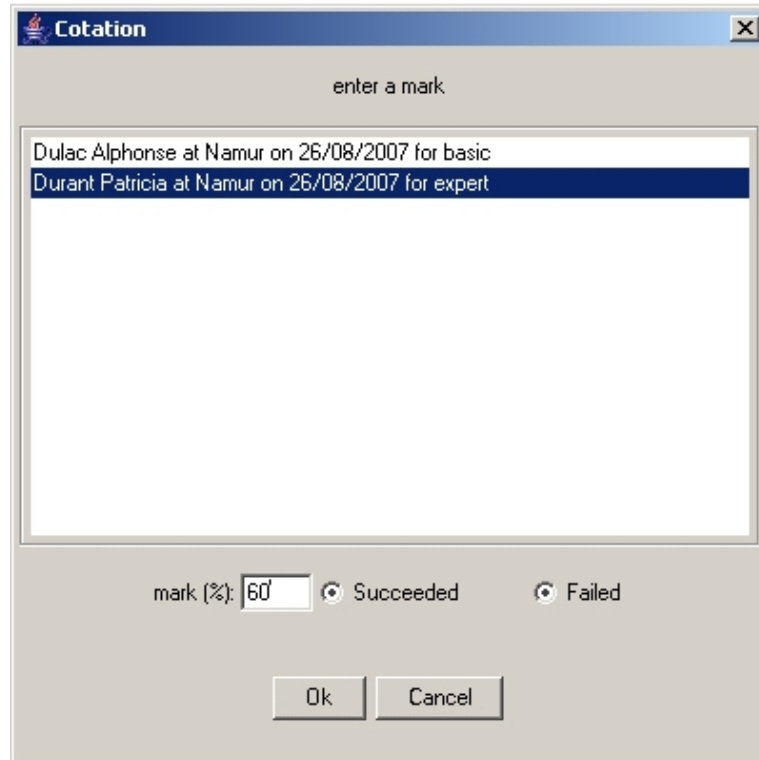


FIG. C.4 – Interface d'attribution d'une note

C.5 Interface de suppression d'un examen

La capture d'écran C.5 liste toutes les dates et lieux d'examination. L'administrateur peut alors sélectionner l'épreuve à supprimer³.

Toutes les interfaces de suppression sont implémentées sur le même principe (i.e. lister, sélectionner, supprimer).

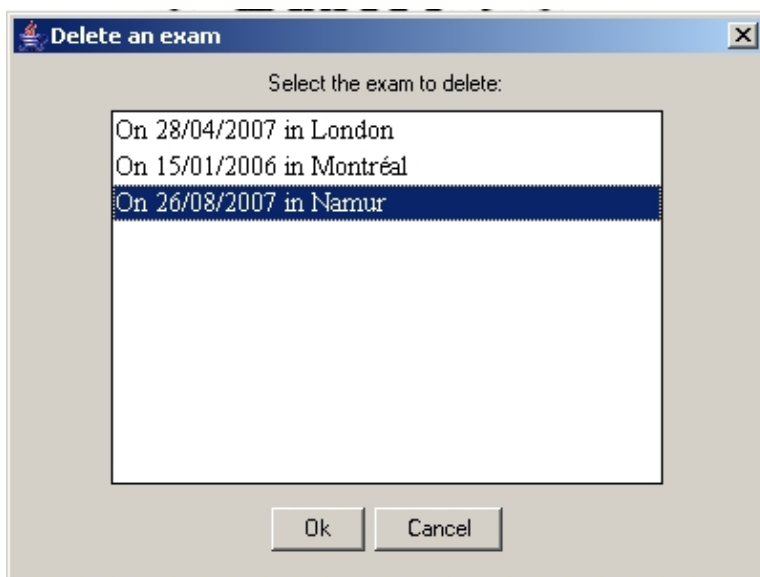


FIG. C.5 – Interface de suppression d'un examen

³Cette action n'est possible que si cet examen ne fait l'objet d'aucune inscription, sans quoi une erreur en informe l'administrateur